FIG.1

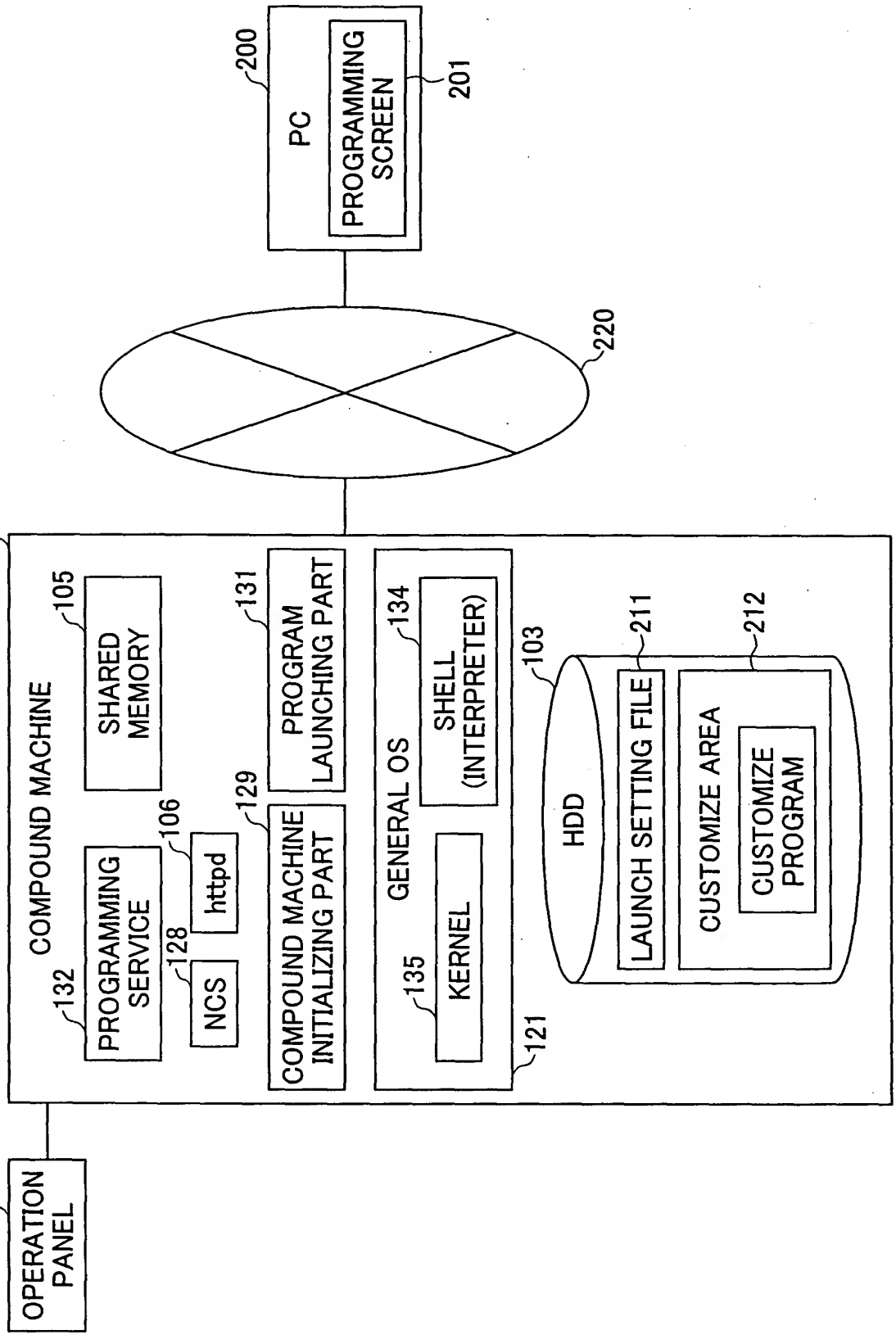

100
COMPOUND MACHINE
105
SHARED MEMORY
131
PROGRAM LAUNCHING PART
132
PROGRAMMING SERVICE
106
128
NCS
httpd
129
COMPOUND MACHINE INITIALIZING PART
GENERAL OS
134
SHELL (INTERPRETER)
135
KERNEL
121
103
HDD
211
LAUNCH SETTING FILE
212
CUSTOMIZE AREA
CUSTOMIZE PROGRAM
210
OPERATION PANEL
200
PC
PROGRAMMING SCREEN
201
220

FIG.2

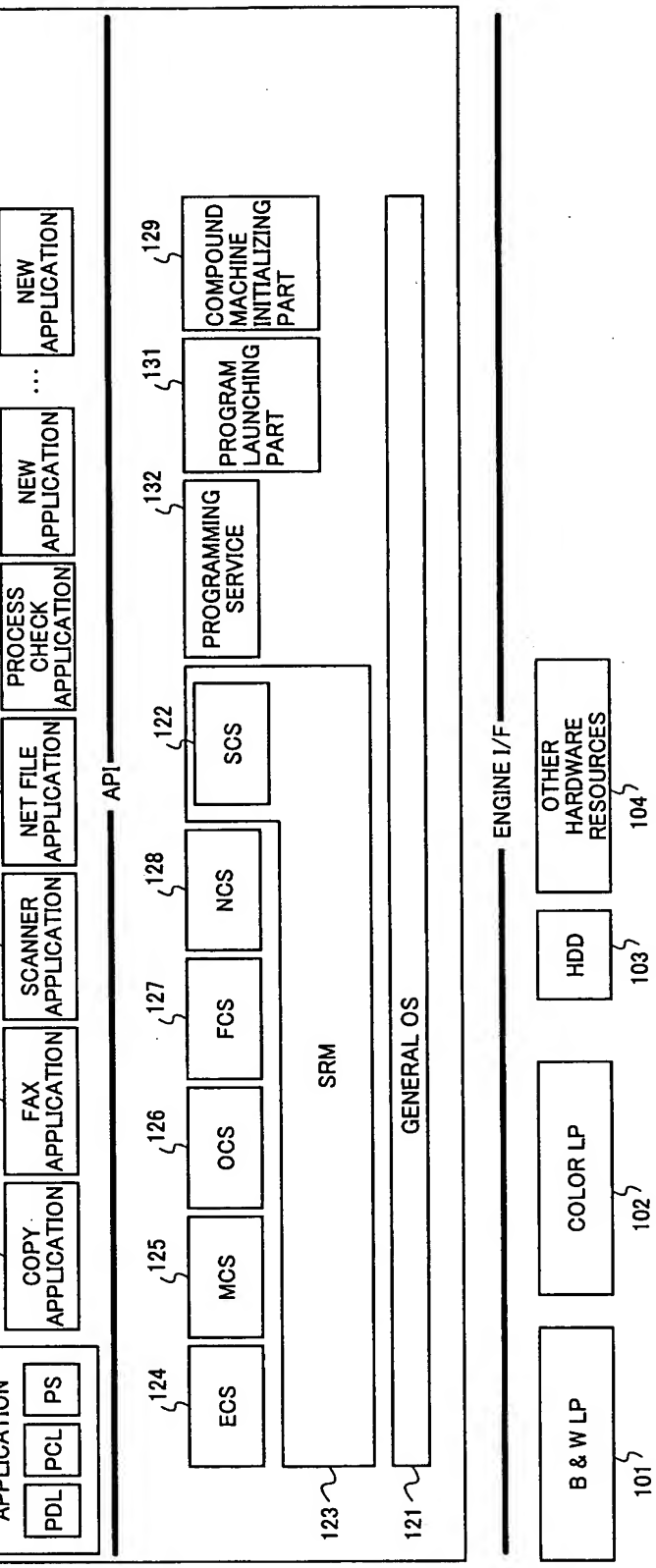

100 COMPOUND MACHINE
110 SOFTWARE GROUP
APPLICATION 130
PLATFORM 120
PRINTER APPLICATION
PDL
PCL
PS
111
COPY APPLICATION
112
FAX APPLICATION
113
SCANNER APPLICATION
114
NET FILE APPLICATION
115
PROCESS CHECK APPLICATION
116
NEW APPLICATION
117
...
NEW APPLICATION
118
API
ECS
124
MCS
125
OCS
126
FCS
127
NCS
128
SRM
123
SCS
122
PROGRAMMING SERVICE
132
PROGRAM LAUNCHING PART
131
COMPOUND MACHINE INITIALIZING PART
129
GENERAL OS
121
ENGINE I/F
B & W LP
101
COLOR LP
102
HDD
103
OTHER HARDWARE RESOURCES
104

FIG.3

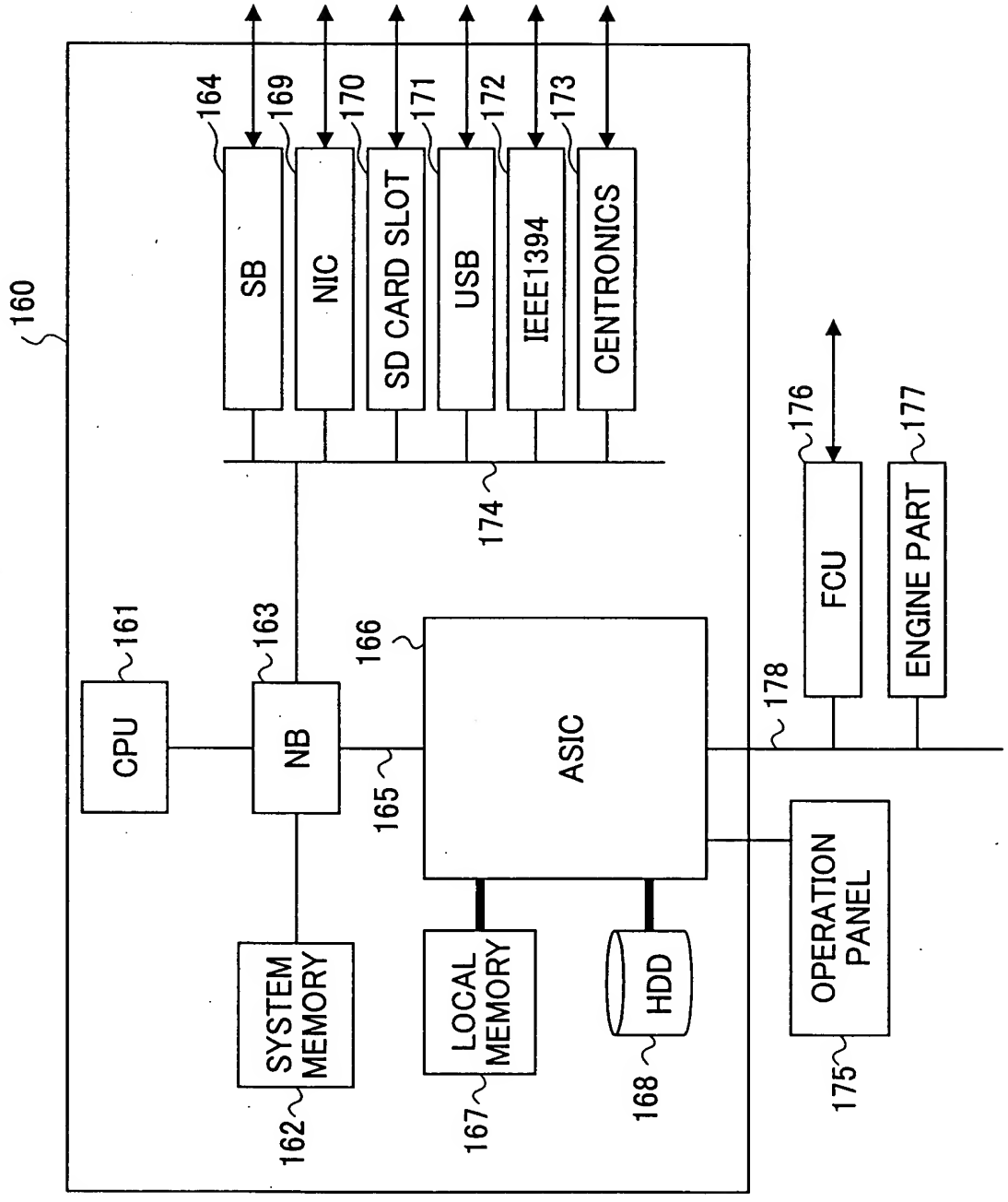

160
164
169
170
171
172
173
SB
NIC
SD CARD SLOT
USB
IEEE1394
CENTRONICS
174
176
177
FCU
ENGINE PART
178
161
163
166
CPU
NB
ASIC
165
SYSTEM MEMORY
LOCAL MEMORY
HDD
OPERATION PANEL
162
167
168
175

FIG.4

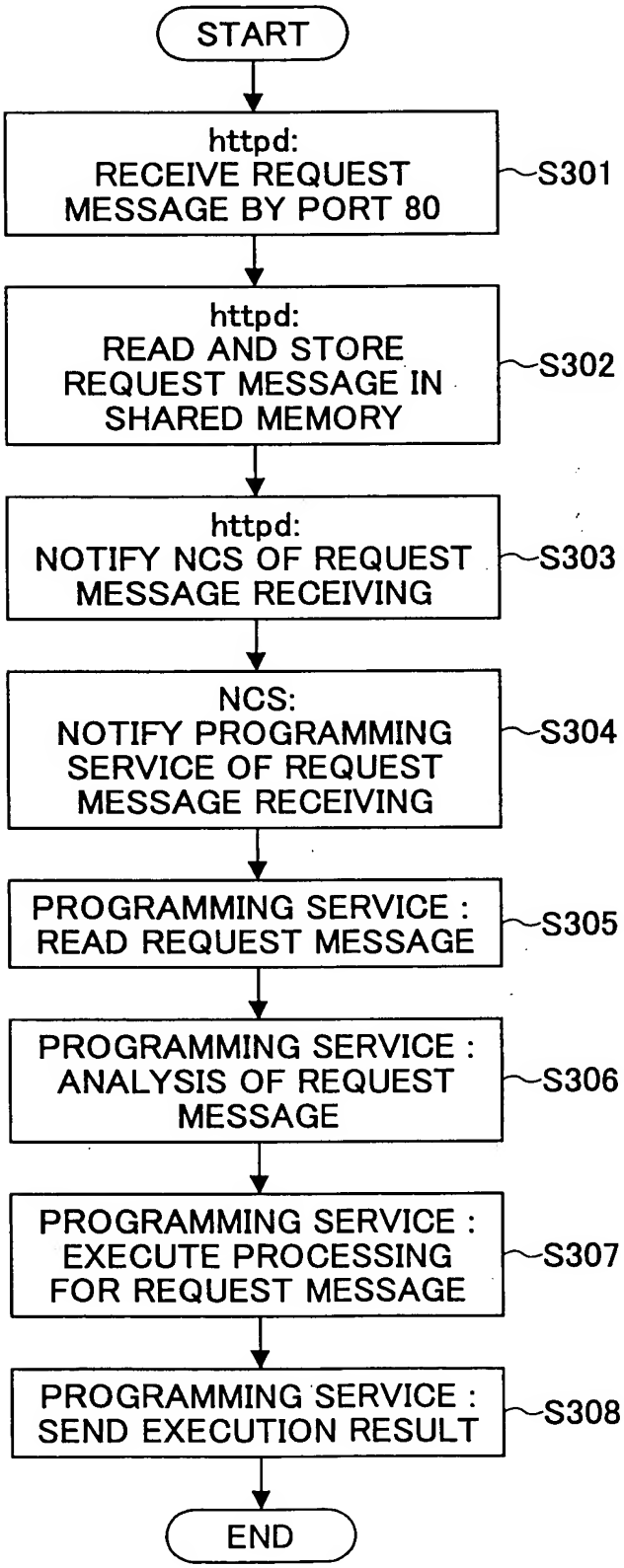

START
httpd: RECEIVE REQUEST MESSAGE BY PORT 80
S301
httpd: READ AND STORE REQUEST MESSAGE IN SHARED MEMORY
S302
httpd: NOTIFY NCS OF REQUEST MESSAGE RECEIVING
S303
NCS: NOTIFY PROGRAMMING SERVICE OF REQUEST MESSAGE RECEIVING
S304
PROGRAMMING SERVICE : READ REQUEST MESSAGE
S305
PROGRAMMING SERVICE : ANALYSIS OF REQUEST MESSAGE
S306
PROGRAMMING SERVICE : EXECUTE PROCESSING FOR REQUEST MESSAGE
S307
PROGRAMMING SERVICE : SEND EXECUTION RESULT
S308
END

FIG.5

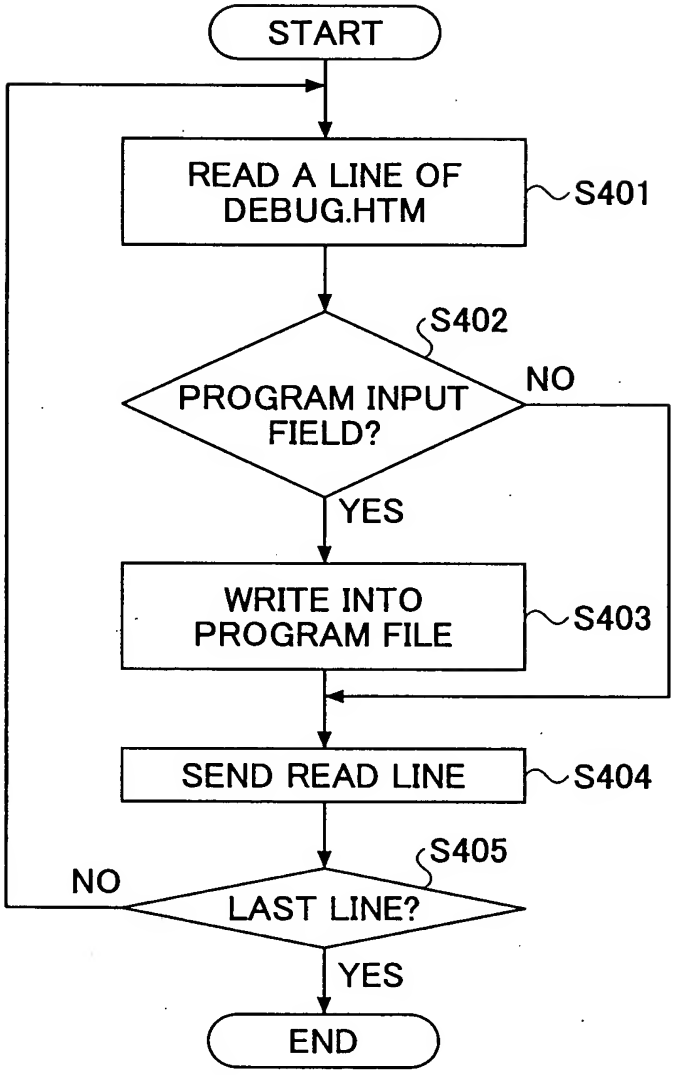

START
READ A LINE OF DEBUG.HTM
S401
S402
PROGRAM INPUT FIELD?
NO
YES
WRITE INTO PROGRAM FILE
S403
SEND READ LINE
S404
S405
NO
LAST LINE?
YES
END

FIG.6

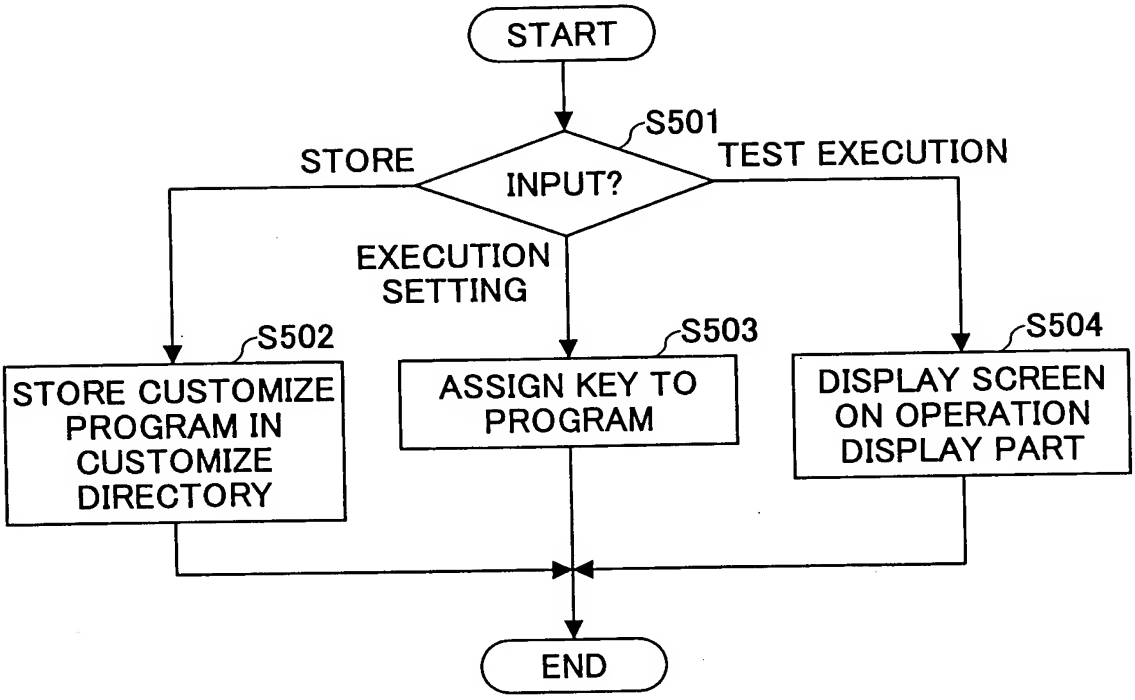

START
S501
INPUT?
STORE
TEST EXECUTION
EXECUTION SETTING
S502
STORE CUSTOMIZE PROGRAM IN CUSTOMIZE DIRECTORY
S503
ASSIGN KEY TO PROGRAM
S504
DISPLAY SCREEN ON OPERATION DISPLAY PART
END

FIG.7

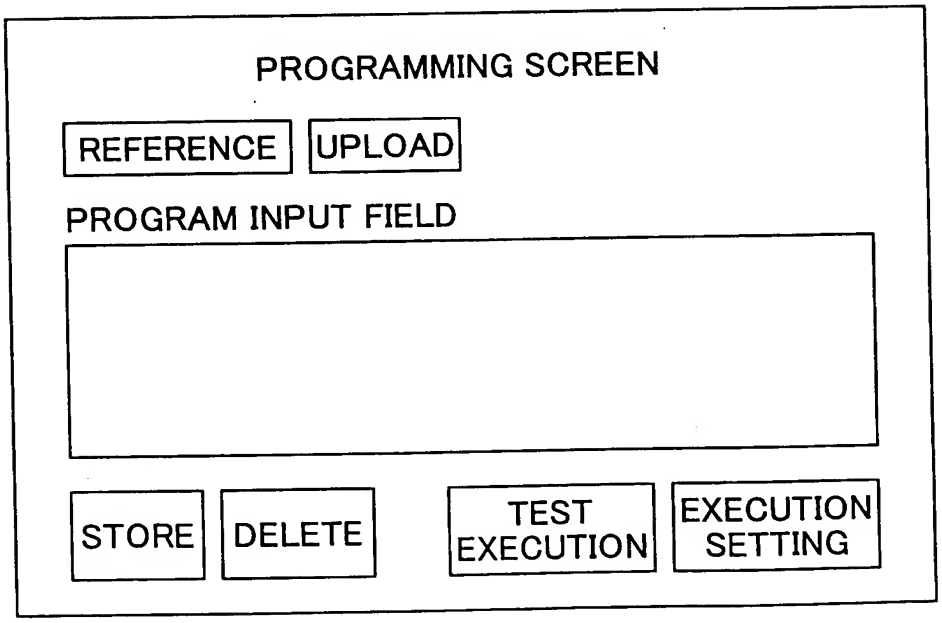

PROGRAMMING SCREEN
REFERENCE
UPLOAD
PROGRAM INPUT FIELD
STORE
DELETE
TEST EXECUTION
EXECUTION SETTING

FIG.8

```
# ocr shell script
#
scanimage -out /hdd/ts/tmp/tmpimg.tif
decomp -in /hdd/ts/tmp/tmpimg.tif -out /hdd/ts/tmp/tmpimg.bmp
ocr -in /hdd/ts/tmp/tmpimg.bmp -out /hdd/ts/tmp/ocrresult.txt
mail xxxxx@yy.zz.co.jp
```

FIG.9

CUSTOMIZE DIRECTORY / FILE NAME

hdd／xxx／opepane／shell1
shell2
shell3

FIG.10

KEY ASSOCIATION TABLE

KEY CODE:	CUSTOMIZE PROGRAM
501 :	shell1
502 :	sh ll2
503 :	sh ll3

FIG.11

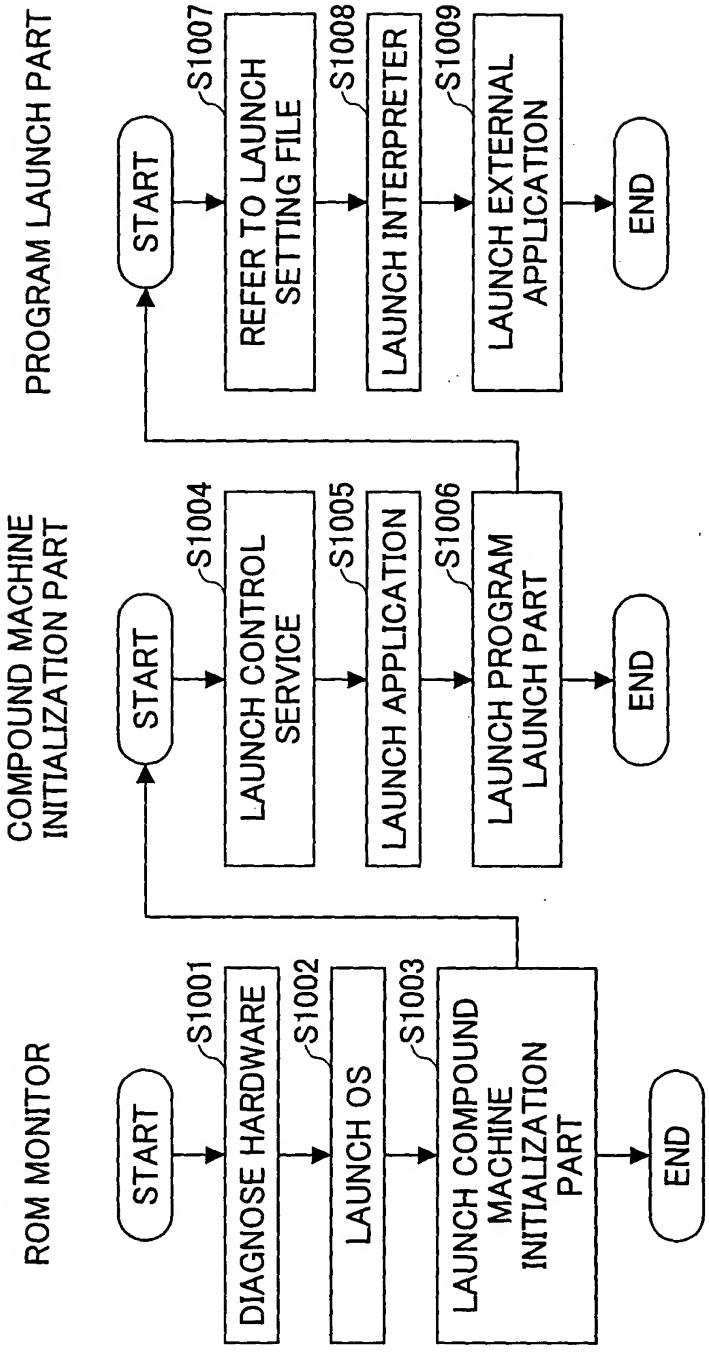

ROM MONITOR
START
S1001
DIAGNOSE HARDWARE
S1002
LAUNCH OS
S1003
LAUNCH COMPOUND MACHINE INITIALIZATION PART
END
COMPOUND MACHINE INITIALIZATION PART
START
S1004
LAUNCH CONTROL SERVICE
S1005
LAUNCH APPLICATION
S1006
LAUNCH PROGRAM LAUNCH PART
END
PROGRAM LAUNCH PART
START
S1007
REFER TO LAUNCH SETTING FILE
S1008
LAUNCH INTERPRETER
S1009
LAUNCH EXTERNAL APPLICATION
END

FIG.12

LAUNCH SETTING FILE

PROGRAM NAME	INTERPRETER (SHELL)
PROGRAM NAME	XXX APPLICATION
⋮	

FIG.13

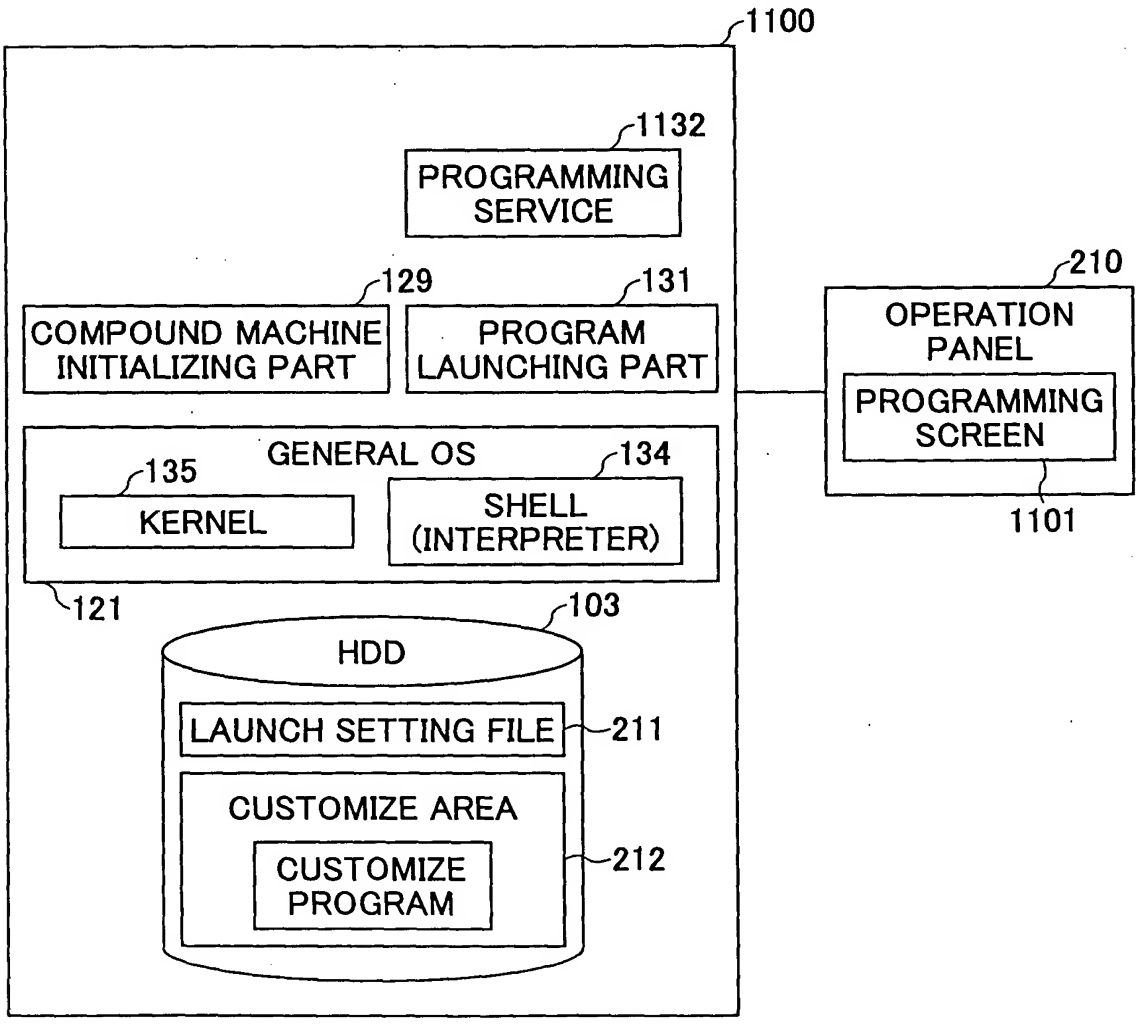


FIG.14

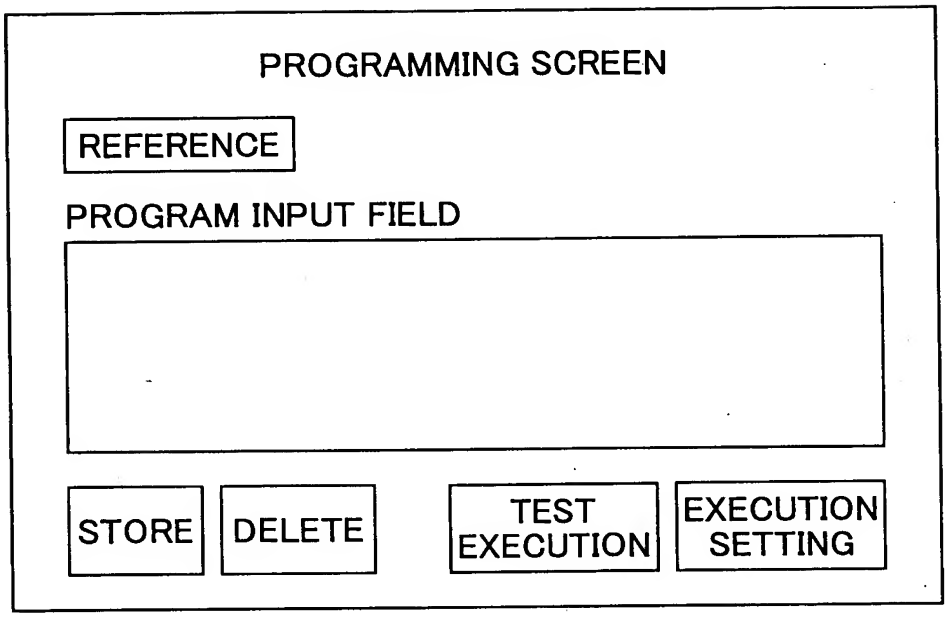

PROGRAMMING SCREEN
REFERENCE
PROGRAM INPUT FIELD
STORE
DELETE
TEST EXECUTION
EXECUTION SETTING

FIG.15

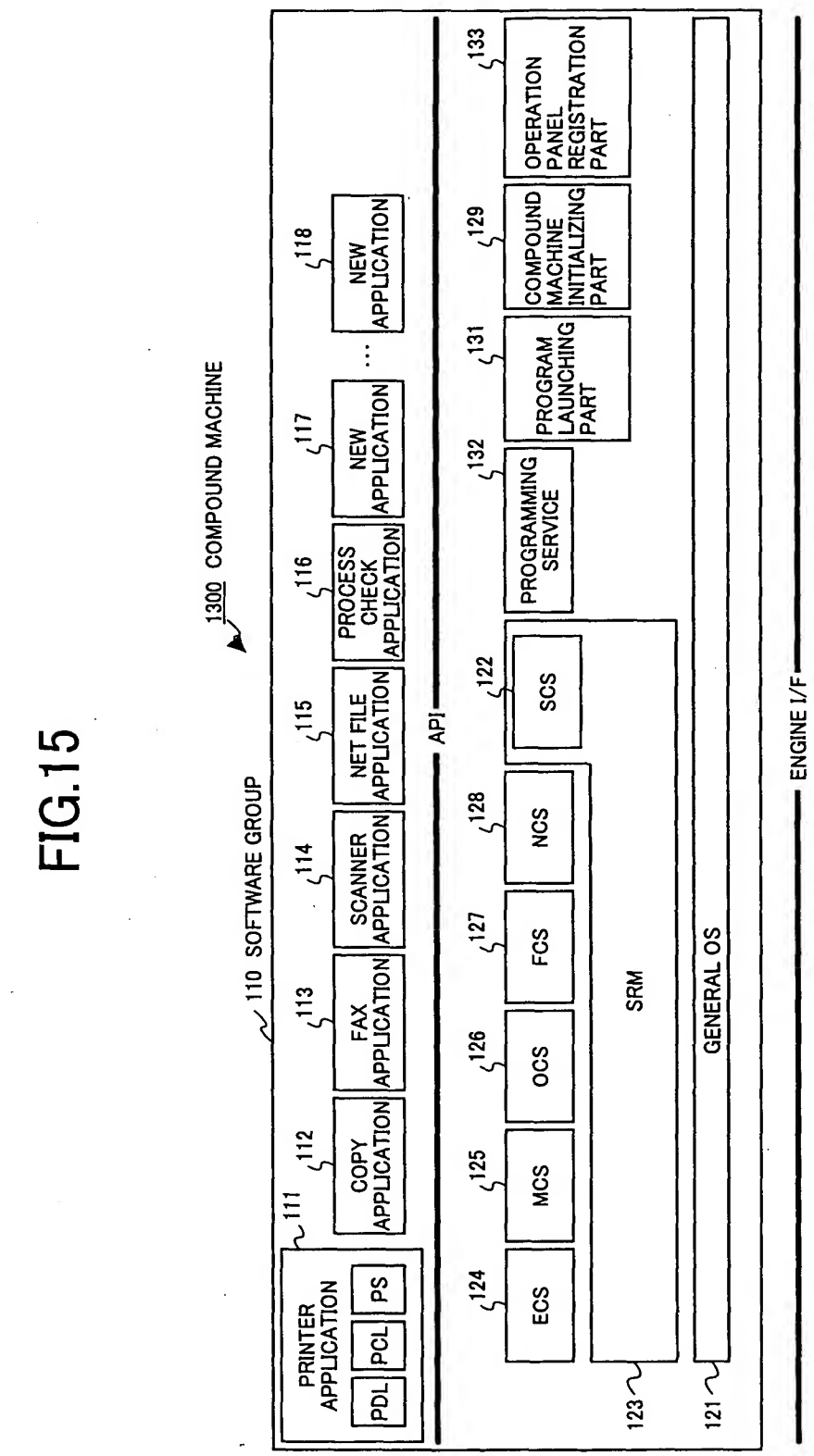

1300 COMPOUND MACHINE
110 SOFTWARE GROUP
PRINTER APPLICATION
PDL
PCL
PS
111
COPY APPLICATION
112
FAX APPLICATION
113
SCANNER APPLICATION
114
NET FILE APPLICATION
115
PROCESS CHECK APPLICATION
116
NEW APPLICATION
117
NEW APPLICATION
118
API
ECS
124
MCS
125
OCS
126
FCS
127
NCS
128
SCS
122
PROGRAMMING SERVICE
132
PROGRAM LAUNCHING PART
131
COMPOUND MACHINE INITIALIZING PART
129
OPERATION PANEL REGISTRATION PART
133
SRM
123
GENERAL OS
121
ENGINE I/F
B & W LP
101
COLOR LP
102
HDD
103
OTHER HARDWARE RESOURCES
104
APPLICATION 130
PLATFORM 120

FIG.16

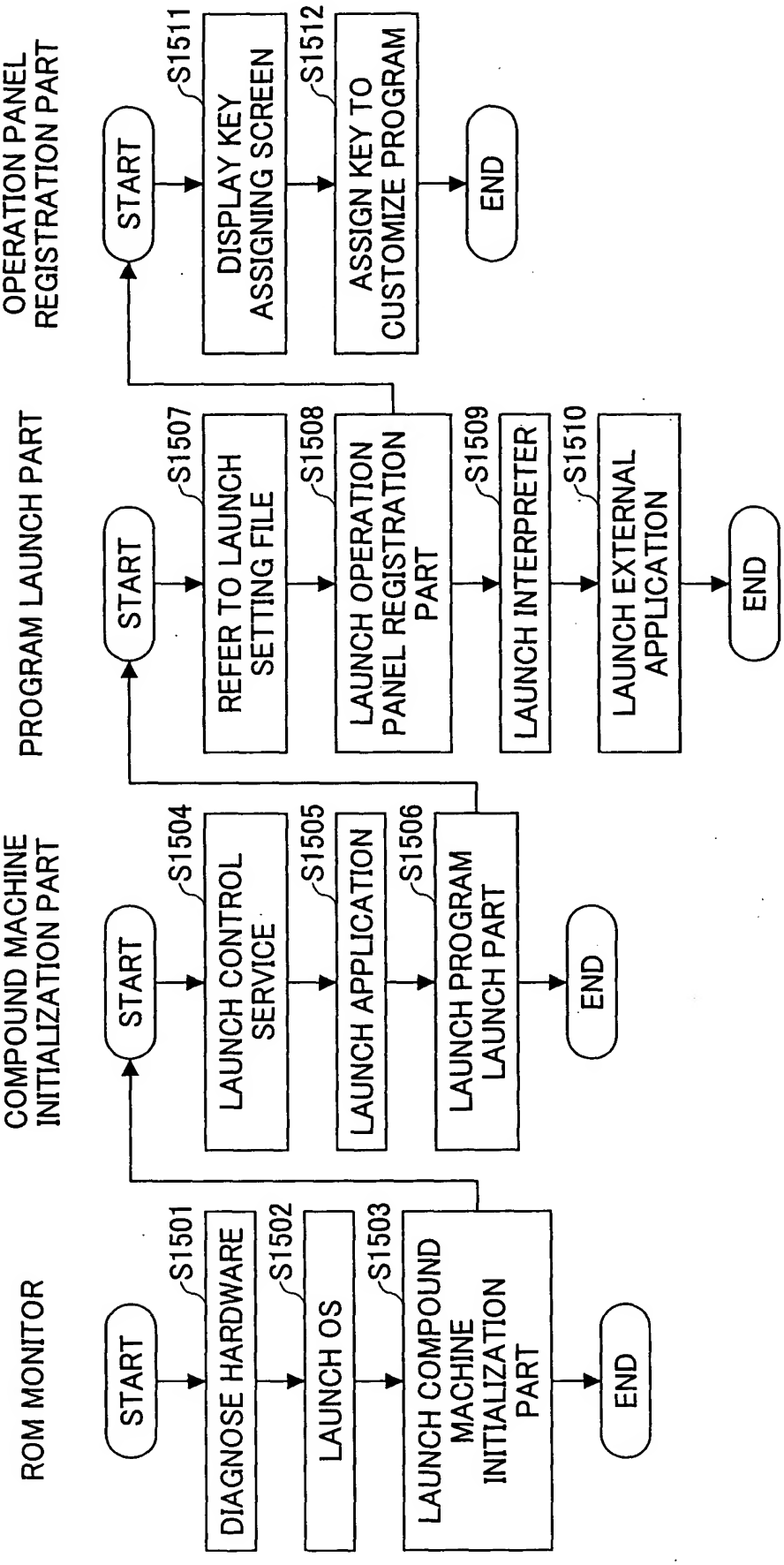

ROM MONITOR
START
S1501
DIAGNOSE HARDWARE
S1502
LAUNCH OS
S1503
LAUNCH COMPOUND MACHINE INITIALIZATION PART
END
COMPOUND MACHINE INITIALIZATION PART
START
S1504
LAUNCH CONTROL SERVICE
S1505
LAUNCH APPLICATION
S1506
LAUNCH PROGRAM LAUNCH PART
END
PROGRAM LAUNCH PART
START
S1507
REFER TO LAUNCH SETTING FILE
S1508
LAUNCH OPERATION PANEL REGISTRATION PART
S1509
LAUNCH INTERPRETER
S1510
LAUNCH EXTERNAL APPLICATION
END
OPERATION PANEL REGISTRATION PART
START
S1511
DISPLAY KEY ASSIGNING SCREEN
S1512
ASSIGN KEY TO CUSTOMIZE PROGRAM
END

FIG.17

LAUNCH SETTING FILE

PROGRAM NAME	OPERATION PANEL REGISTRATION PROGRAM
PROGRAM NAME	INTERPRETER (SHELL)
PROGRAM NAME	XXX APPLICATION
	⋮

FIG.18

KEY ASSIGNMENT SETTING SCREEN

CUSTOMIZE PROGRAM	KEY (BUTTON)		
Shell1	A	B	C
Shell2	A	B	C
Shell3	A	B	C

FIG.19

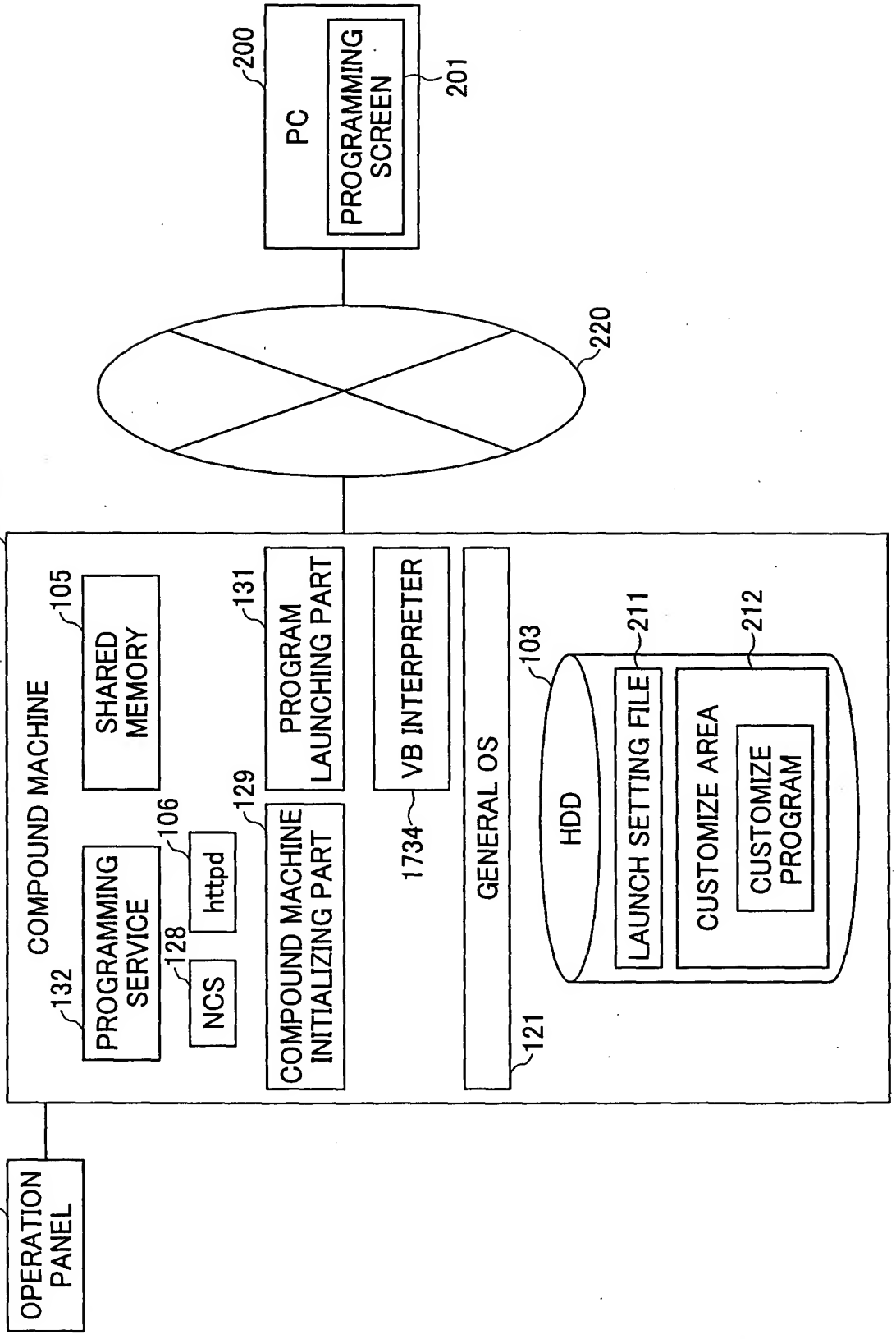

200
PC
PROGRAMMING SCREEN
201
220
1700
COMPOUND MACHINE
105
SHARED MEMORY
131
PROGRAM LAUNCHING PART
VB INTERPRETER
GENERAL OS
103
HDD
211
LAUNCH SETTING FILE
212
CUSTOMIZE AREA
CUSTOMIZE PROGRAM
106
PROGRAMMING SERVICE
132
128
NCS
httpd
129
COMPOUND MACHINE INITIALIZING PART
1734
121
210
OPERATION PANEL

FIG.20

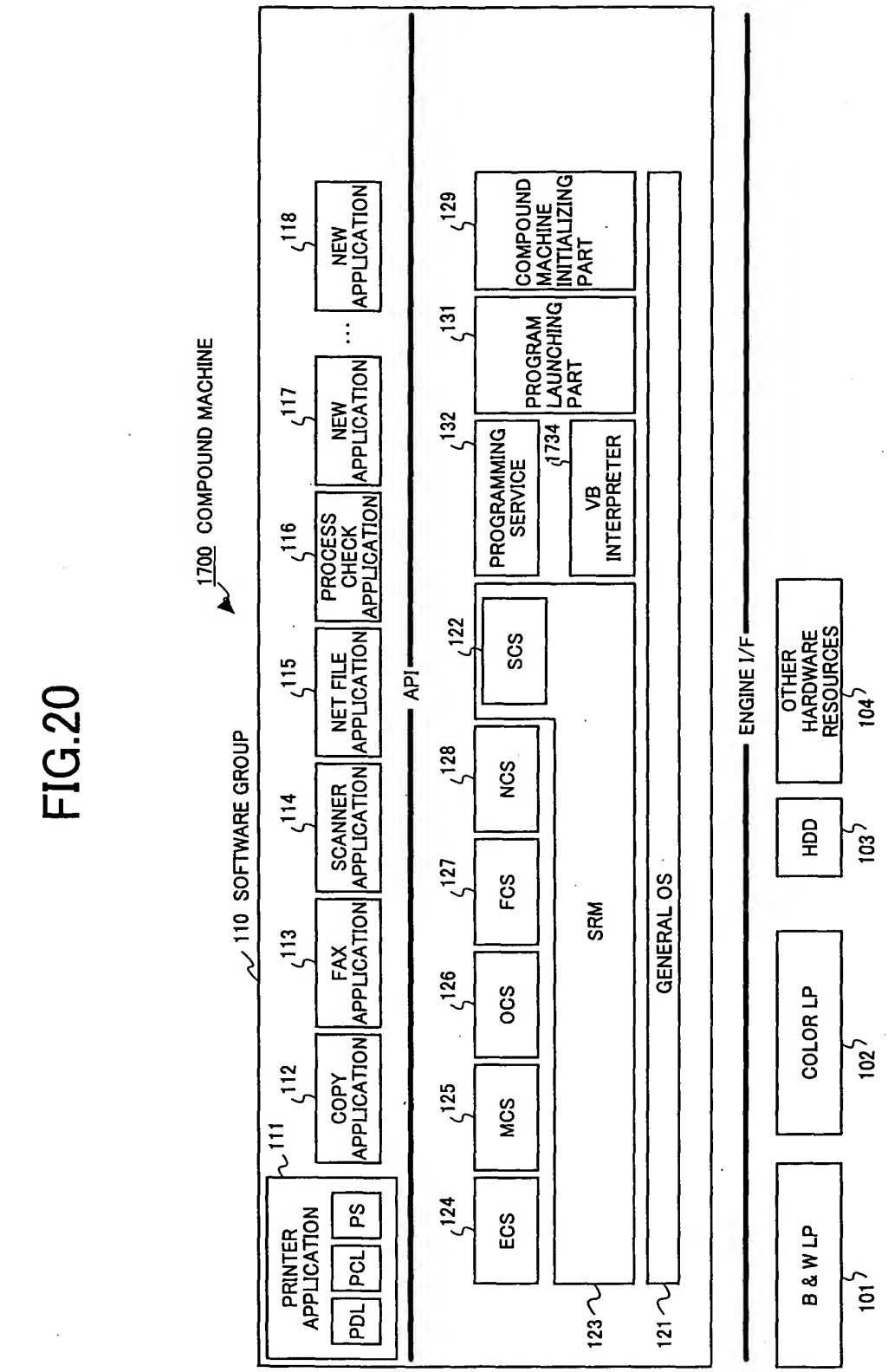

1700 COMPOUND MACHINE
110 SOFTWARE GROUP
PRINTER APPLICATION
PDL
PCL
PS
111
COPY APPLICATION
112
FAX APPLICATION
113
SCANNER APPLICATION
114
NET FILE APPLICATION
115
PROCESS CHECK APPLICATION
116
NEW APPLICATION
117
NEW APPLICATION
118
API
ECS
124
MCS
125
OCS
126
FCS
127
NCS
128
SCS
122
PROGRAMMING SERVICE
132
VB INTERPRETER
1734
PROGRAM LAUNCHING PART
131
COMPOUND MACHINE INITIALIZING PART
129
SRM
123
GENERAL OS
121
APPLICATION 130
PLATFORM 120
ENGINE I/F
B & W LP
101
COLOR LP
102
HDD
103
OTHER HARDWARE RESOURCES
104

FIG.21

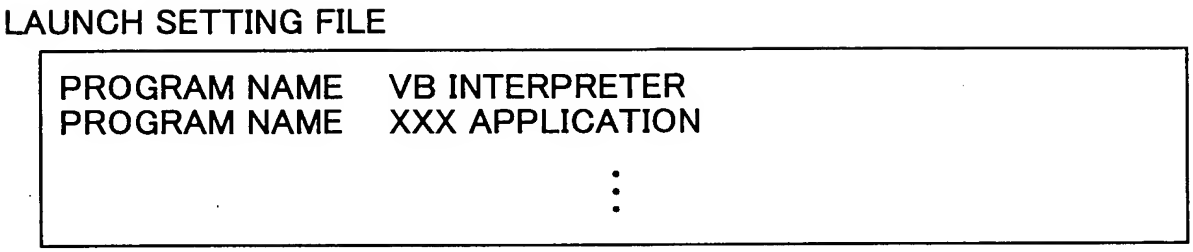

LAUNCH SETTING FILE
PROGRAM NAME VB INTERPRETER
PROGRAM NAME XXX APPLICATION

FIG.22

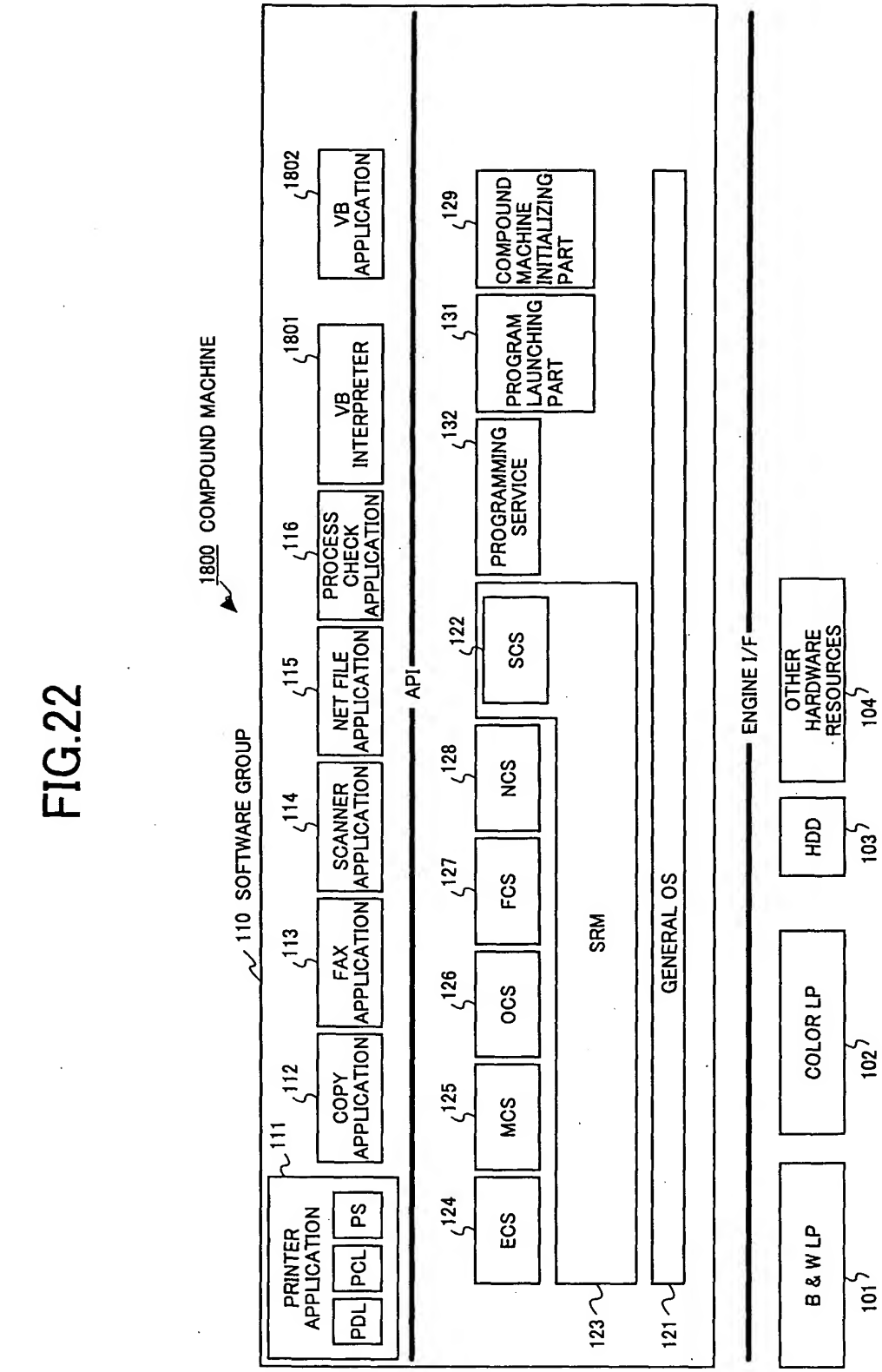

1800 COMPOUND MACHINE
110 SOFTWARE GROUP
PRINTER APPLICATION
PDL
PCL
PS
111
COPY APPLICATION
112
FAX APPLICATION
113
SCANNER APPLICATION
114
NET FILE APPLICATION
115
PROCESS CHECK APPLICATION
116
VB INTERPRETER
1801
VB APPLICATION
1802
API
ECS
124
MCS
125
OCS
126
FCS
127
NCS
128
SCS
122
PROGRAMMING SERVICE
132
PROGRAM LAUNCHING PART
131
COMPOUND MACHINE INITIALIZING PART
129
SRM
123
GENERAL OS
121
ENGINE I/F
B & W LP
101
COLOR LP
102
HDD
103
OTHER HARDWARE RESOURCES
104
APPLICATION 130
PLATFORM 120

FIG.23

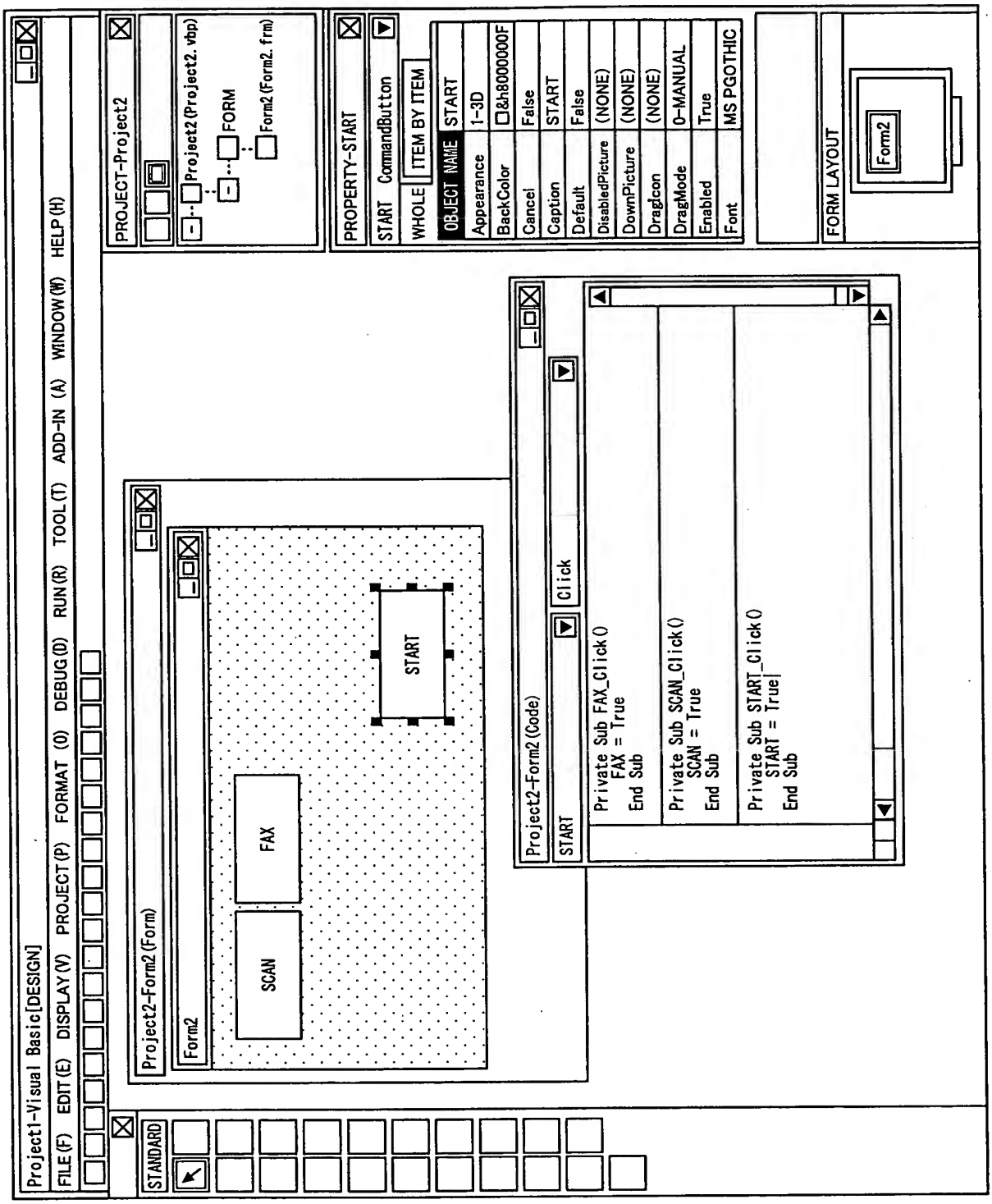

Project1-Visual Basic[DESIGN]
FILE(F) EDIT(E) DISPLAY(V) PROJECT(P) FORMAT(O) DEBUG(D) RUN(R) TOOL(T) ADD-IN(A) WINDOW(W) HELP(H)
STANDARD
PROJECT-Project2
Project2(Project2.vbp)
FORM
Form2(Form2.frm)
PROPERTY-START
START CommandButton
WHOLE ITEM BY ITEM
OBJECT NAME START
Appearance 1-3D
BackColor &h8000000F
Cancel False
Caption START
Default False
DisabledPicture (NONE)
DownPicture (NONE)
DragIcon (NONE)
DragMode 0-MANUAL
Enabled True
Font MS PGOTHIC
FORM LAYOUT
Form2
Project2-Form2(Form)
Form2
SCAN
FAX
START
Project2-Form2(Code)
START
Click
Private Sub FAX_Click()
FAX = True
End Sub
Private Sub SCAN_Click()
SCAN = True
End Sub
Private Sub START_Click()
START = True
End Sub

FIG.24

```
Begin VB. Form Form1
    Caption          =  "Form1"
    ClientHeight     =  3390
    ClientLeft       =  60
    ClientTop        =  345
    ClientWidth      =  6990
    LinkTopic        =  "Form1"
    ScaleHeight      =  3390
    ScaleWight       =  6990
    StartUpPosition  =  3 'Windows DEFAULT
    Begin VB. CommandButton NEXTWIN
        Caption      =  "NEXTWIN"
        Height       =  735
        Left         =  480
        TabIndex     =  0
        Top          =  480
        Width        =  1575
    End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False

Private Sub NEXTWIN_Click()
    /*Form2 SWITCH TO FORM 2 WINDOW */
    call("change_diplay", "From2")
End Sub
```

FIG.25

```
Begin VB. Form Form2
      Caption              =  "Form2"
      ClientHeight         =  3390
      ClientLeft           =  60
      ClientTop            =  345
      ClientWidth          =  6990
      LinkTopic            =  "Form2"
      ScaleHeight          =  3390
      ScaleWight           =  6990
      StartUpPosition      =  3 'Windows DEFAULT
      Begin VB. CommandButton START
            Caption              =  "START"
            Height               =  735
            Left                 =  4560
            TabIndex             =  2
            Top                  =  2040
            Width                =  1815
      End
      Begin VB. CommandButton FAX
            Caption              =  "FAX"
            Height               =  735
            Left                 =  2520
            TabIndex             =  1
            Top                  =  480
            Width                =  1455
      End
      Begin VB. CommandButton SCAN
            Caption              =  "SCAN"
            Height               =  735
            Left                 =  480
            TabIndex             =  0
            Top                  =  480
            Width                =  1575
      End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False

Private Sub FAX_Click()
    FAX = True
End Sub
Private Sub SCAN_Click()
    call("ScanStart", A4, ADF, TIFF, BINARY, "/work/tmpfile.tif")
    gwOpItemCreate(win, MESSAGE_ITEM,
                   ITEM_MESSAGE_X,          0,
                   ITEM_MESSAGE_Y,          12,
                   ITEM_MESSAGE_WIDTH,      100,
                   ITEM_MESSAGE_HEIGHT,     12,
                   ITEM_MESSAGE_CENTER_X,   0,
                   ITEM_MESSAGE_BLINK,      0,
                   ITEM_MESSAGE,            "CHARACTER STRING", 0,
                   ITEM_MESSAGE_FRONT,      FONT_12, 0,
                   0) ;
End Sub

Private Sub START_Click()
    START = True
End Sub
```

FIG.26

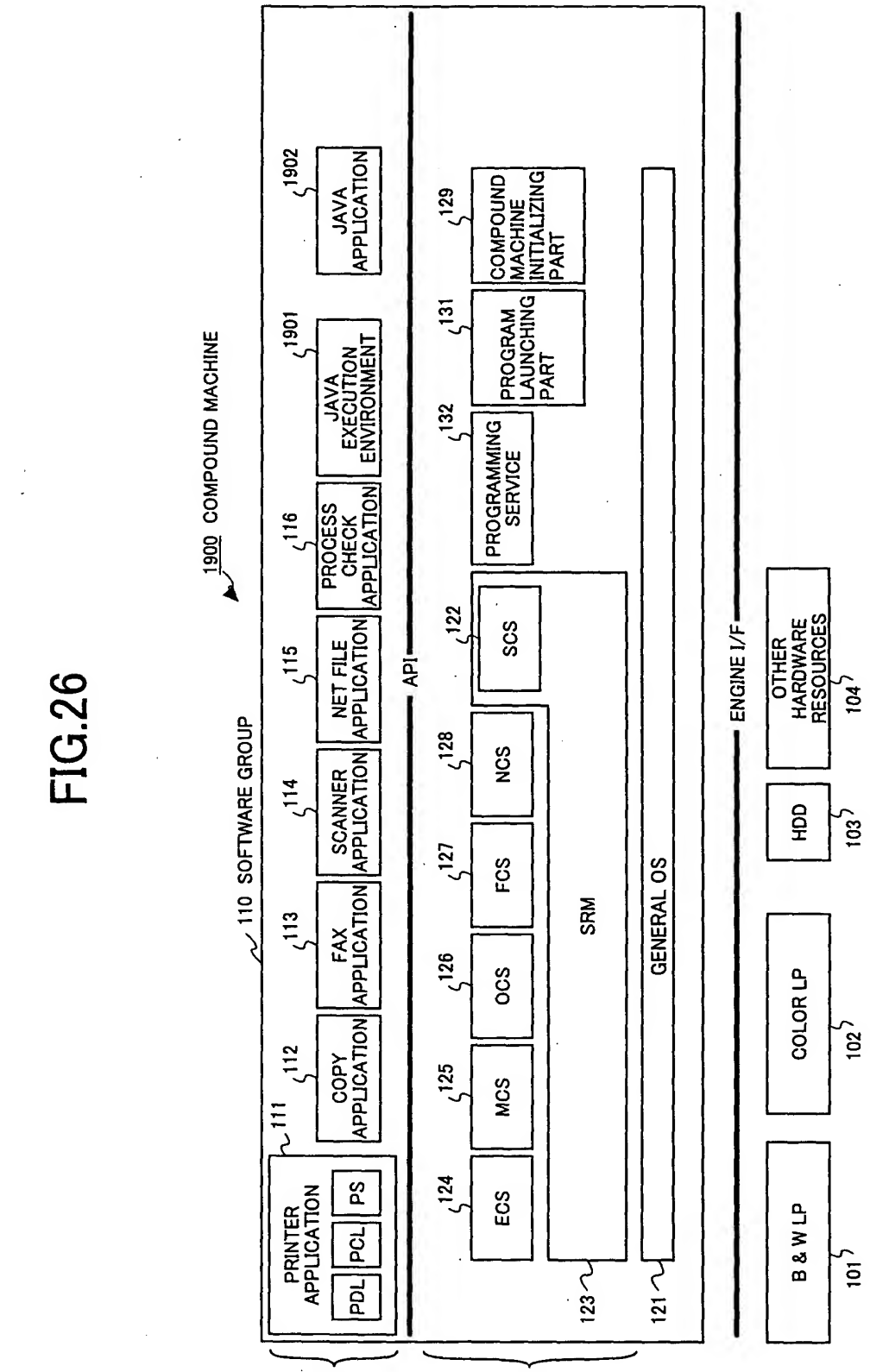

1900 COMPOUND MACHINE
110 SOFTWARE GROUP
PRINTER APPLICATION
PDL
PCL
PS
111
COPY APPLICATION
112
FAX APPLICATION
113
SCANNER APPLICATION
114
NET FILE APPLICATION
115
PROCESS CHECK APPLICATION
116
JAVA EXECUTION ENVIRONMENT
1901
JAVA APPLICATION
1902
API
ECS
124
MCS
125
OCS
126
FCS
127
NCS
128
SCS
122
PROGRAMMING SERVICE
132
PROGRAM LAUNCHING PART
131
COMPOUND MACHINE INITIALIZING PART
129
SRM
123
GENERAL OS
121
ENGINE I/F
B & W LP
101
COLOR LP
102
HDD
103
OTHER HARDWARE RESOURCES
104
APPLICATION 130
PLATFORM 120

FIG.27

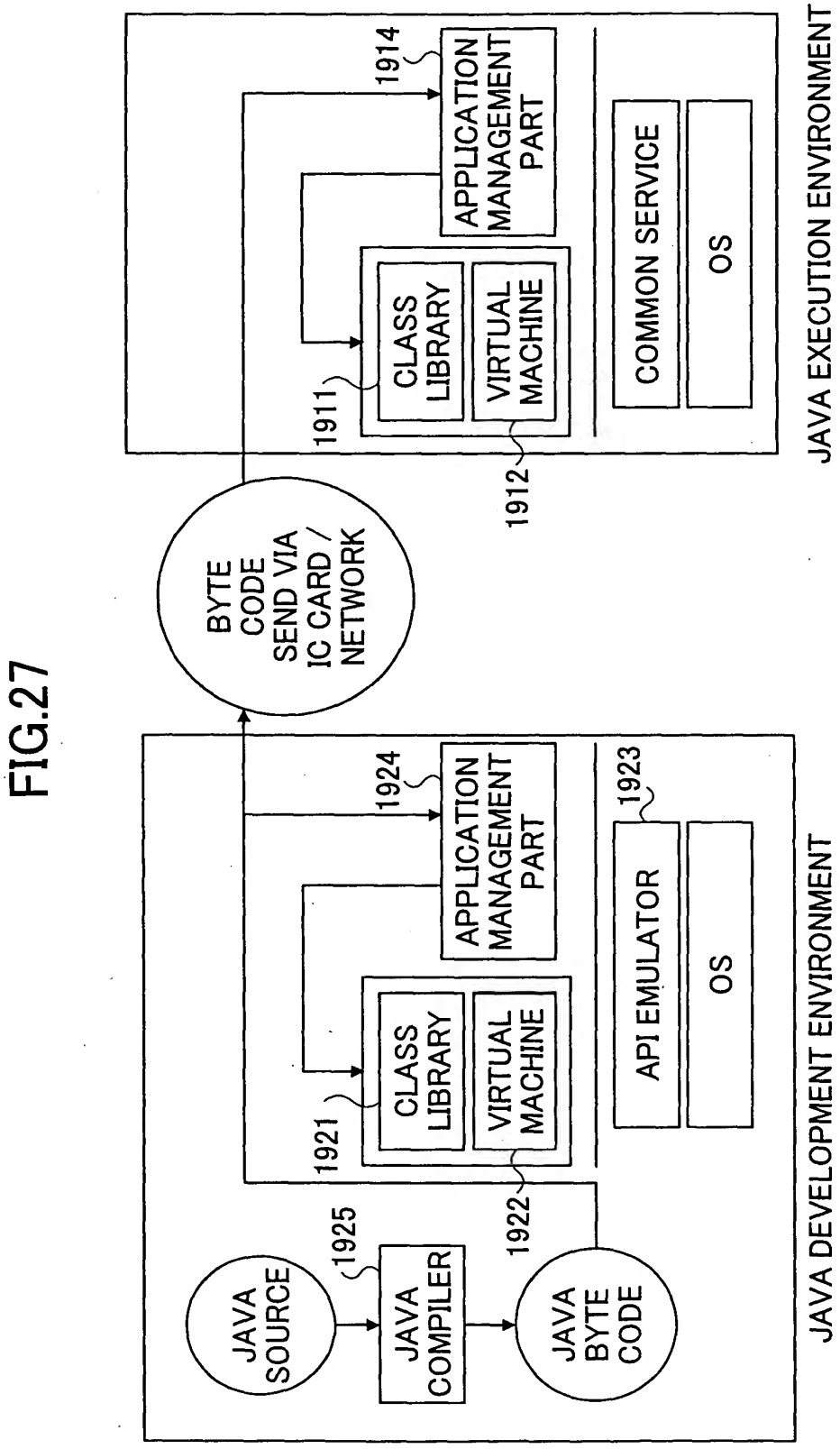

JAVA DEVELOPMENT ENVIRONMENT
JAVA SOURCE
JAVA COMPILER
JAVA BYTE CODE
1925
1922
1921
CLASS LIBRARY
VIRTUAL MACHINE
1924
APPLICATION MANAGEMENT PART
API EMULATOR
1923
OS
BYTE CODE SEND VIA IC CARD / NETWORK
1912
1911
CLASS LIBRARY
VIRTUAL MACHINE
1914
APPLICATION MANAGEMENT PART
COMMON SERVICE
OS
JAVA EXECUTION ENVIRONMENT

FIG.28

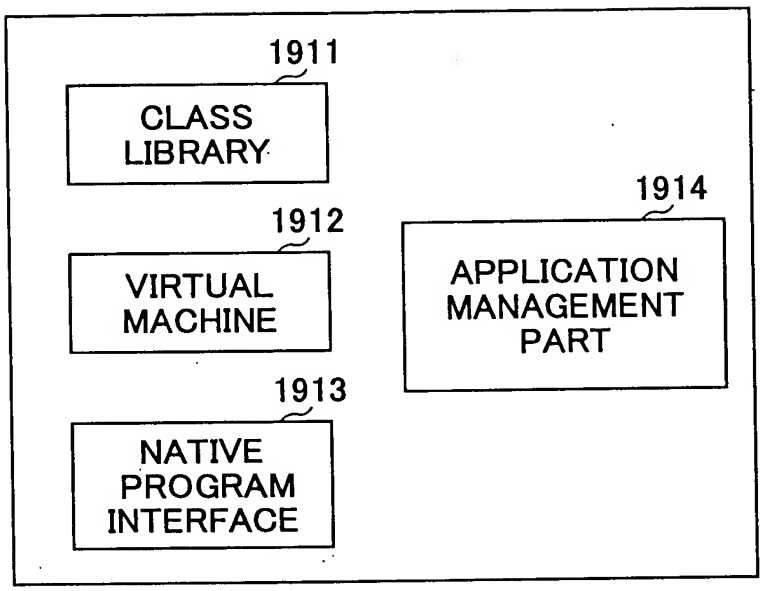

1911
CLASS LIBRARY
1912
VIRTUAL MACHINE
1913
NATIVE PROGRAM INTERFACE
1914
APPLICATION MANAGEMENT PART

FIG.29

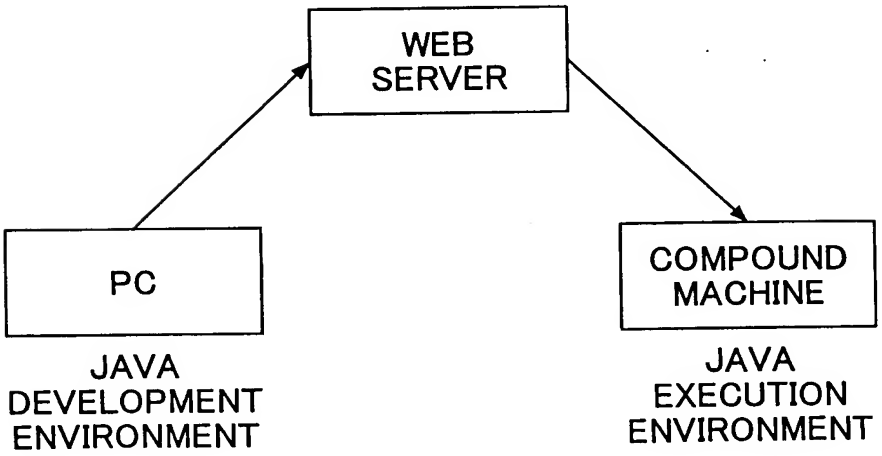

WEB SERVER
PC
JAVA DEVELOPMENT ENVIRONMENT
COMPOUND MACHINE
JAVA EXECUTION ENVIRONMENT

FIG.30

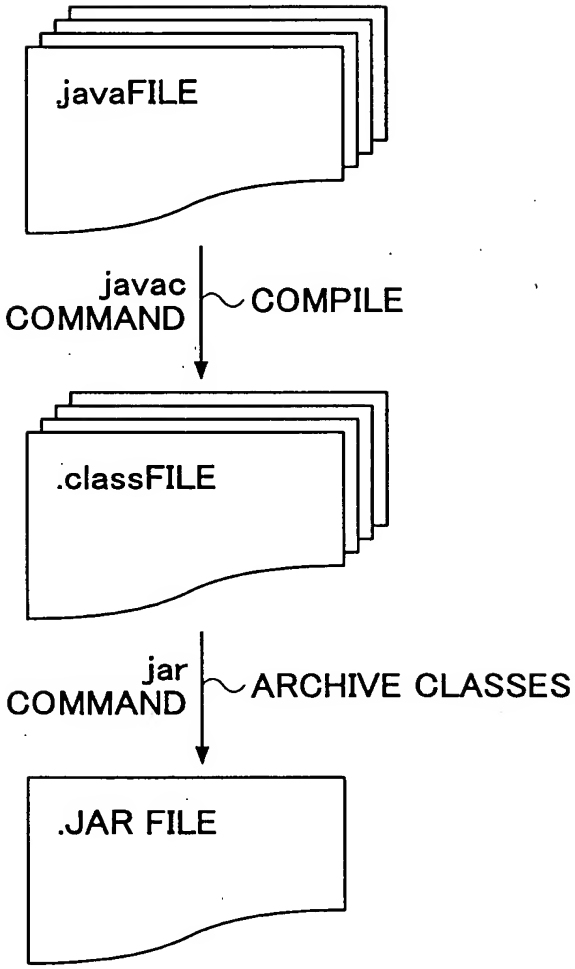

javaFILE
javac
COMMAND
COMPILE
.classFILE
jar
COMMAND
ARCHIVE CLASSES
.JAR FILE

FIG.31

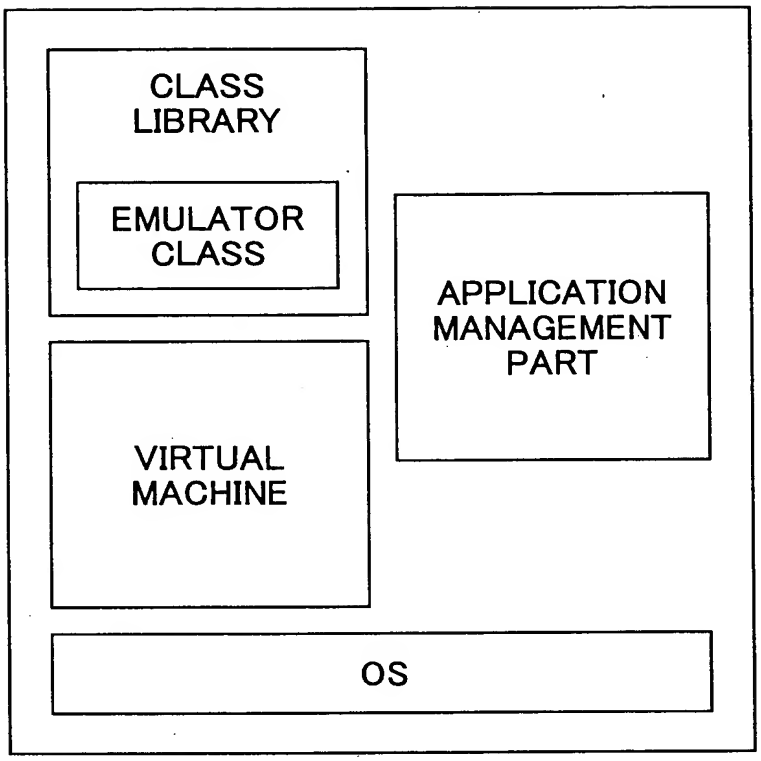

CLASS LIBRARY
EMULATOR CLASS
APPLICATION MANAGEMENT PART
VIRTUAL MACHINE
OS
JAVA DEVELOPMENT ENVIRONMENT

FIG.32

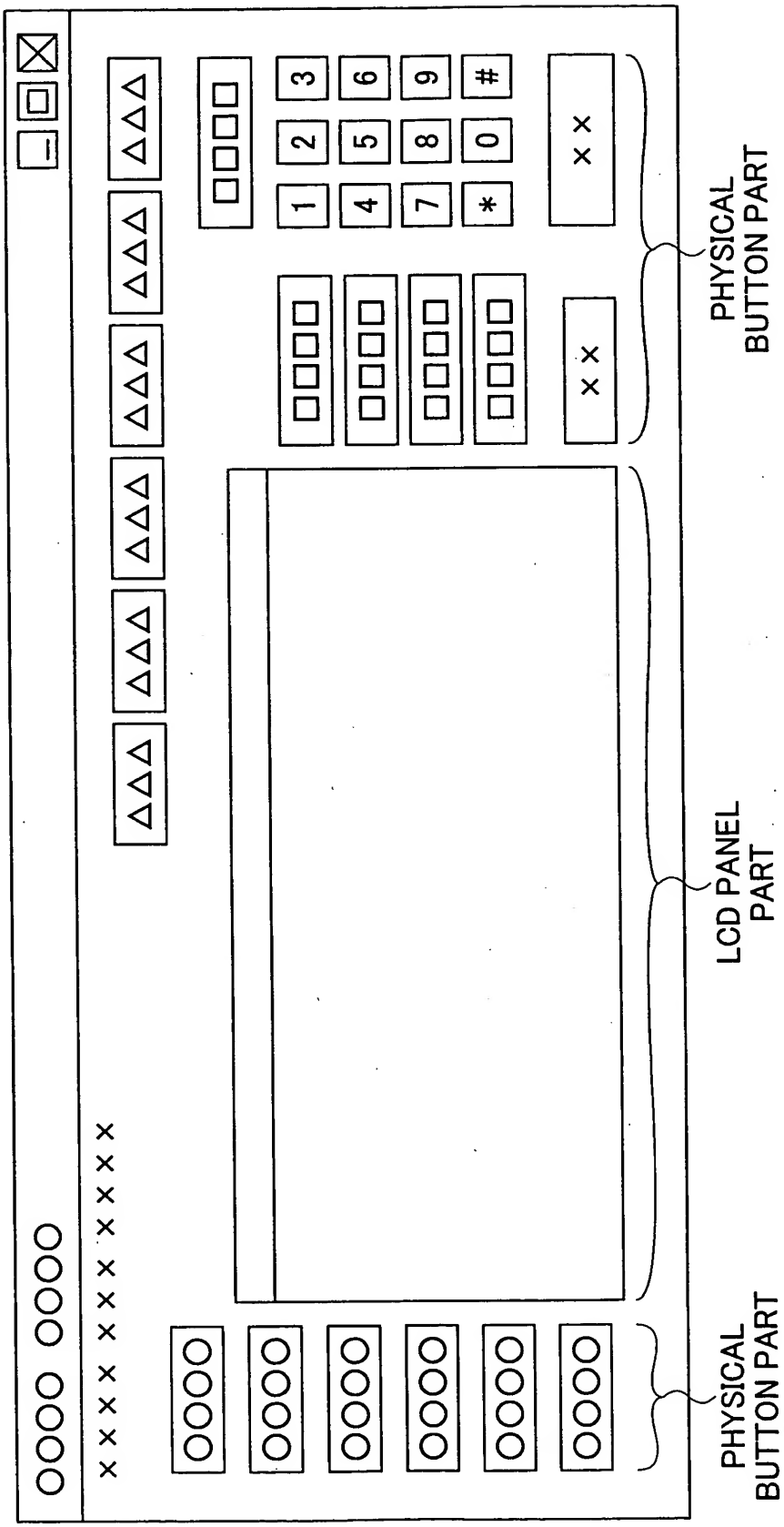


FIG.33

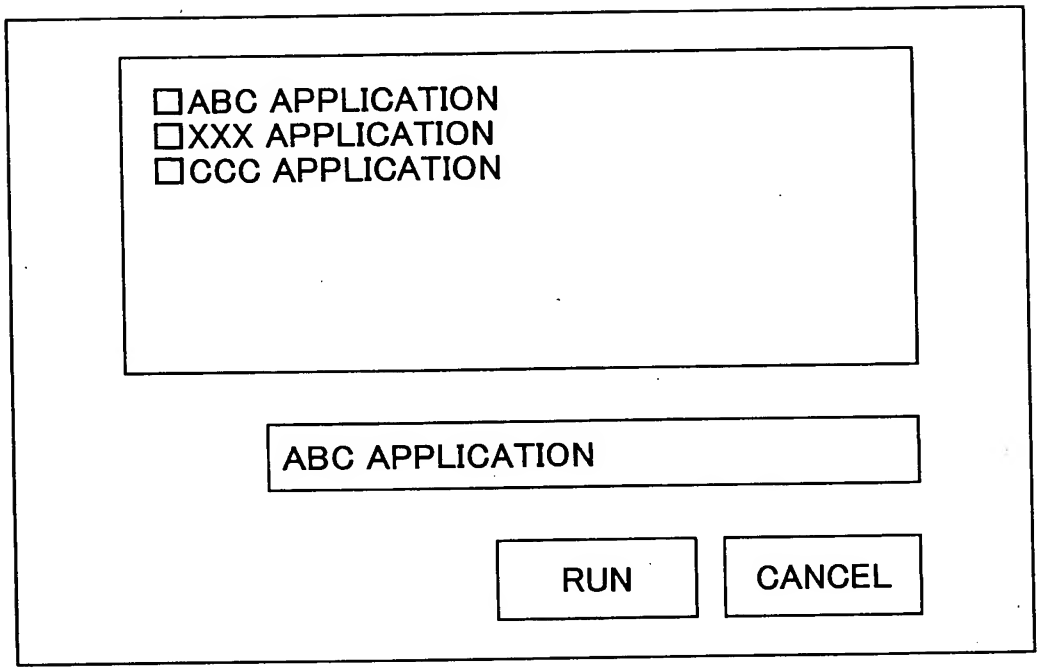

☐ABC APPLICATION
☐XXX APPLICATION
☐CCC APPLICATION
ABC APPLICATION
RUN
CANCEL

FIG.34

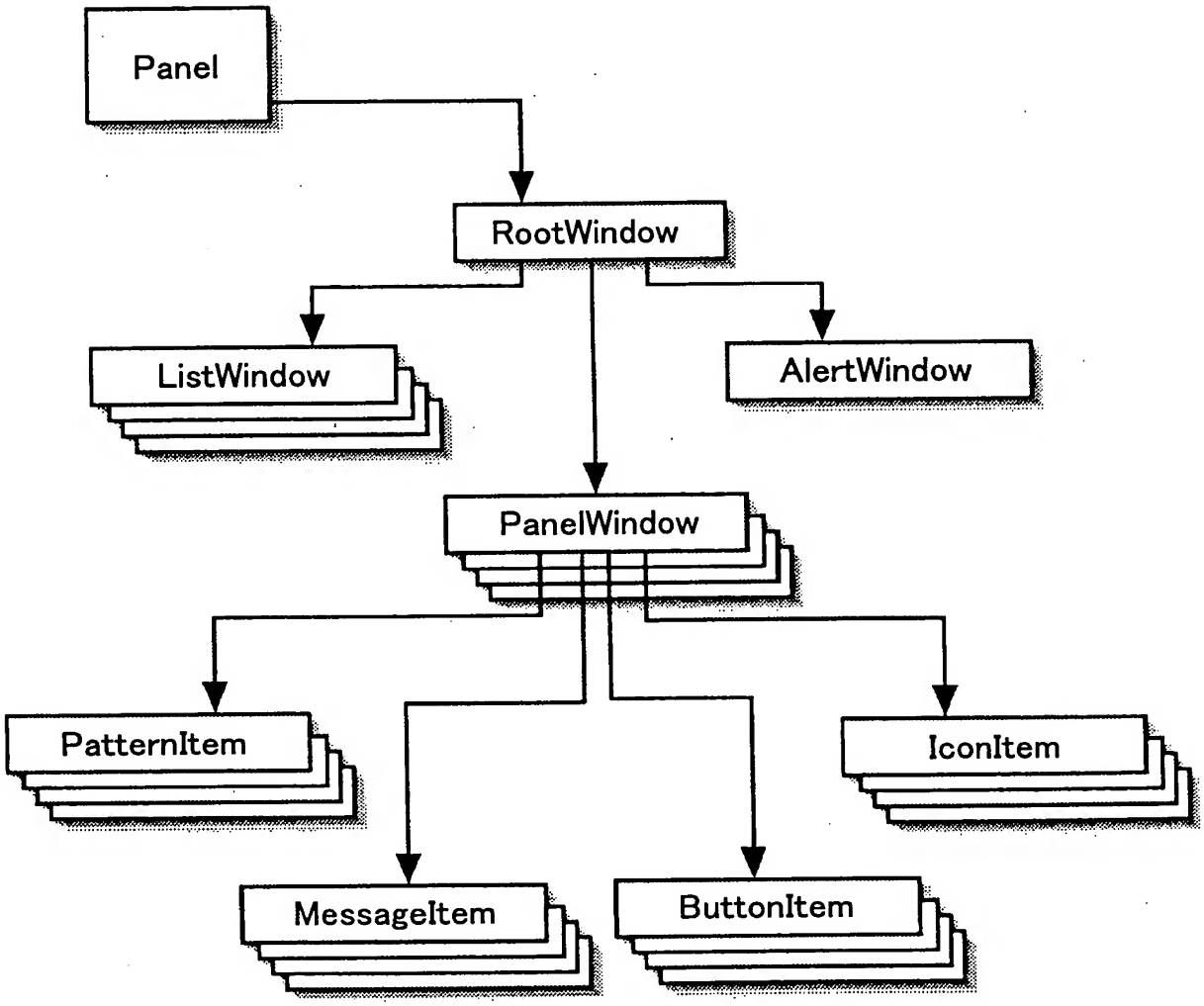

Panel
RootWindow
ListWindow
AlertWindow
PanelWindow
PatternItem
IconItem
MessageItem
ButtonItem

FIG.35

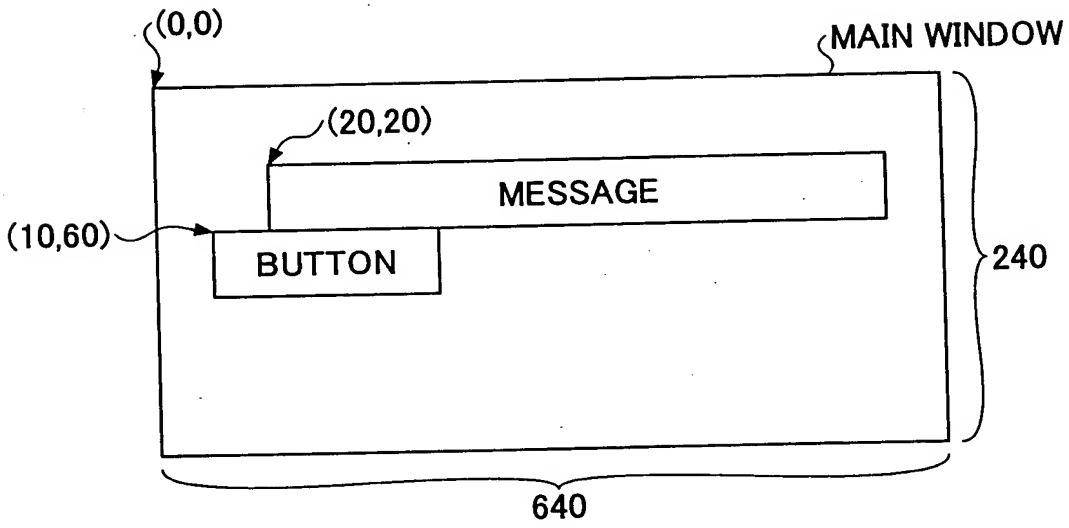


FIG.36

```
import java.lang.*;
import com.ricoh.gw.panel.*;                                    //←①
public class UserApp extends GWApp                              //←②
{
    public void main()
    {
        Panel panel = new Panel();
        PanelWindow mainwindow =
        new PanelWindow(panel.root());
        ItemString hello = new ItemString("Hello World");       //←③
        MessageItem salutation = new MessageItem(hello);
        salutation.setRect(100,10,100,10);
        mainwindow.addItem(salutation);
        ItemString inputstr =
        new ItemString("Get input...");
        ButtonItem inputButton = new ButtonItem();              //←④
        inputButton.setButtonShape(ButtonItem.BTN_ZAB);
        inputButton.setRect(100,100,100,100);
        inputButton.setWink(true);
        inputButton.addChangeListener(
            new GWChangeListener()
            {
                public void stateChanged(GWEvent e)
                {
                    String answer =
                        SoftKeyboardWindow.request(
                            "Add your Input",
                            "", 256,
                            ItemString.LANG_ENGLISH_US
                        );
                    inputdisplay.setMessage(
                            new ItemString(answer));
                    inputdisplay.paint();
                }
            }
        );
        mainwindow.addItem(inputButton);
        ItemString label1 =
            new ItemString("Your Input Was...");                //←⑤
        MessageItem label = new MessageItem(label1);
        label.setRect(100,110,200,10);
        mainwindow.addItem(label);
        ItemString outputstr = new ItemString ("");
        MessageItem
        inputdisplay = new MessageItem(outputstr);              //←⑥
        inputdisplay.setRect(100,125,200,10);
        mainwindow.addItem(inputdisplay);
    }
}
```

FIG.37

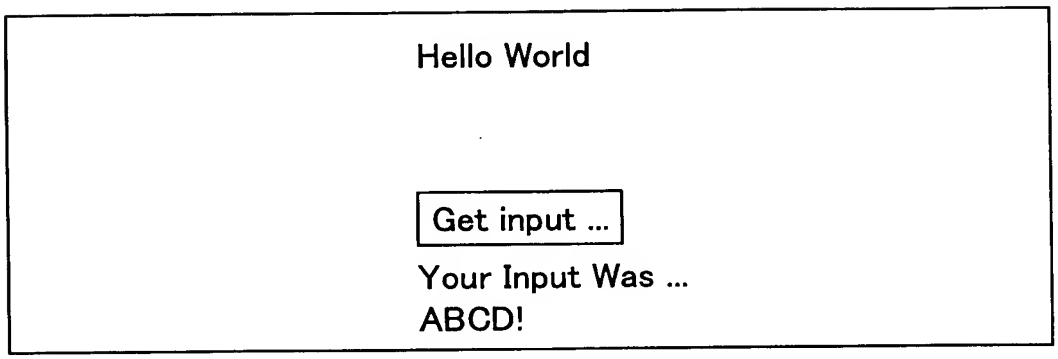

Hello World
Get input ...
Your Input Was ...
ABCD!

FIG.38

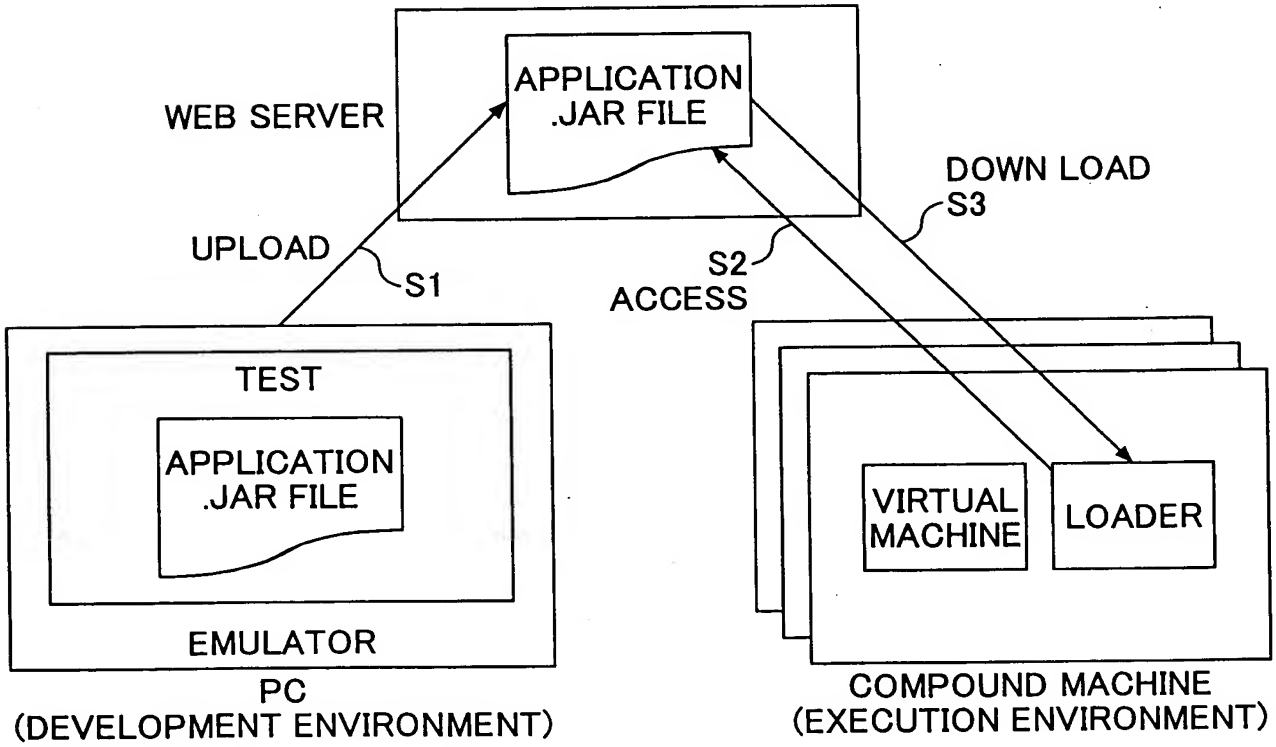

WEB SERVER
APPLICATION .JAR FILE
DOWN LOAD
S3
UPLOAD
S1
S2
ACCESS
TEST
APPLICATION .JAR FILE
EMULATOR
PC
(DEVELOPMENT ENVIRONMENT)
VIRTUAL MACHINE
LOADER
COMPOUND MACHINE
(EXECUTION ENVIRONMENT)

FIG.39

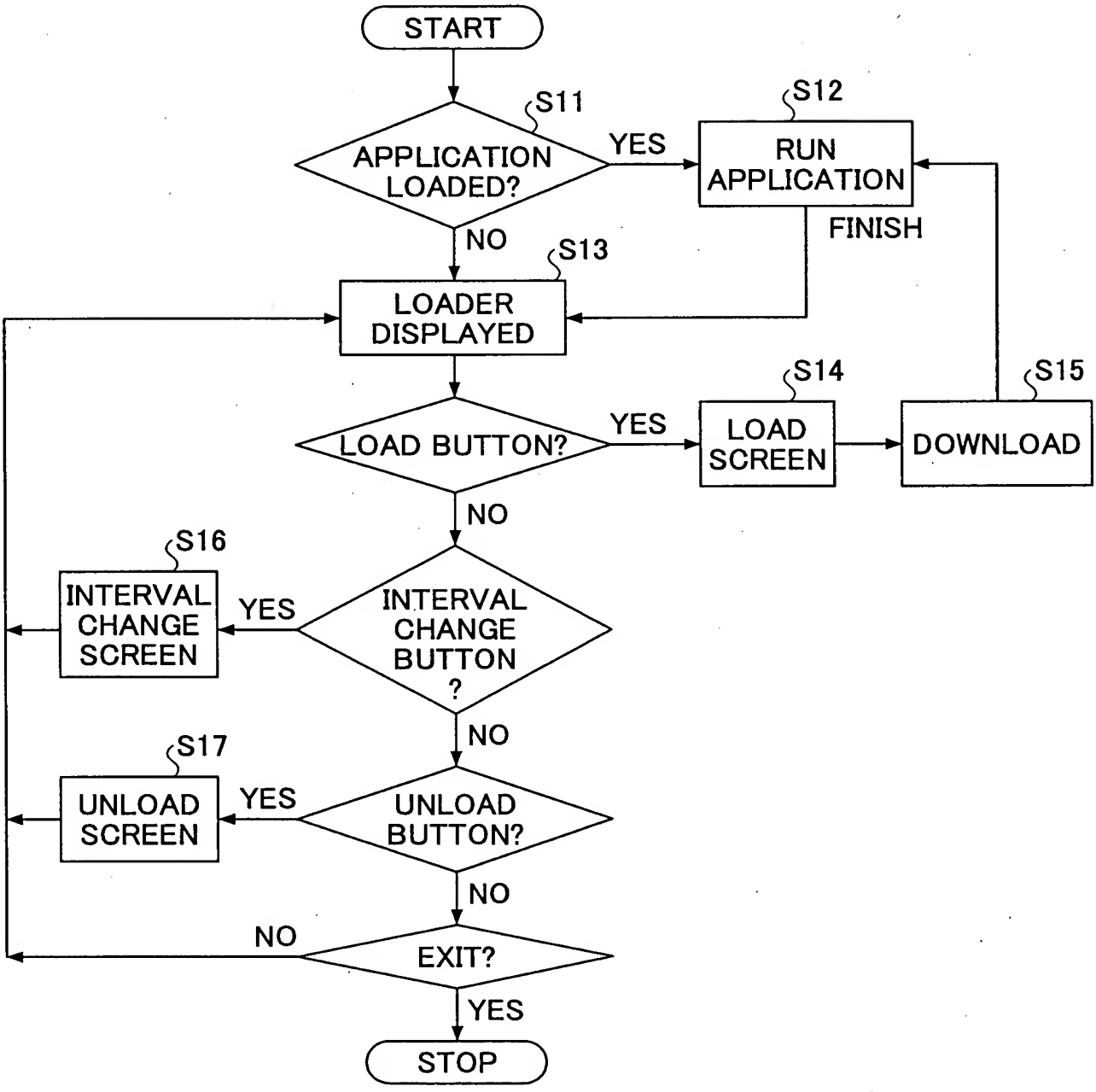

START
S11
APPLICATION LOADED?
YES
S12
RUN APPLICATION
FINISH
NO
S13
LOADER DISPLAYED
S14
LOAD BUTTON?
YES
LOAD SCREEN
S15
DOWNLOAD
NO
S16
INTERVAL CHANGE SCREEN
YES
INTERVAL CHANGE BUTTON ?
NO
S17
UNLOAD SCREEN
YES
UNLOAD BUTTON?
NO
NO
EXIT?
YES
STOP

FIG.40

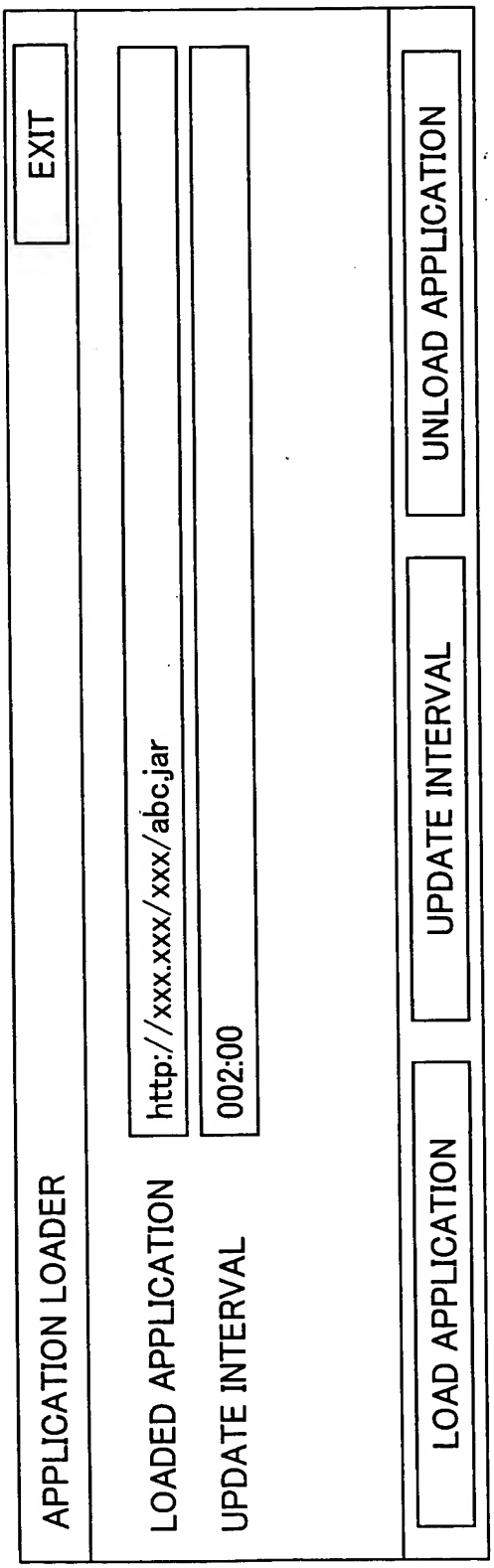

APPLICATION LOADER
EXIT
LOADED APPLICATION
http://xxx.xxx/xxx/abc.jar
UPDATE INTERVAL
002:00
LOAD APPLICATION
UPDATE INTERVAL
UNLOAD APPLICATION

FIG.41

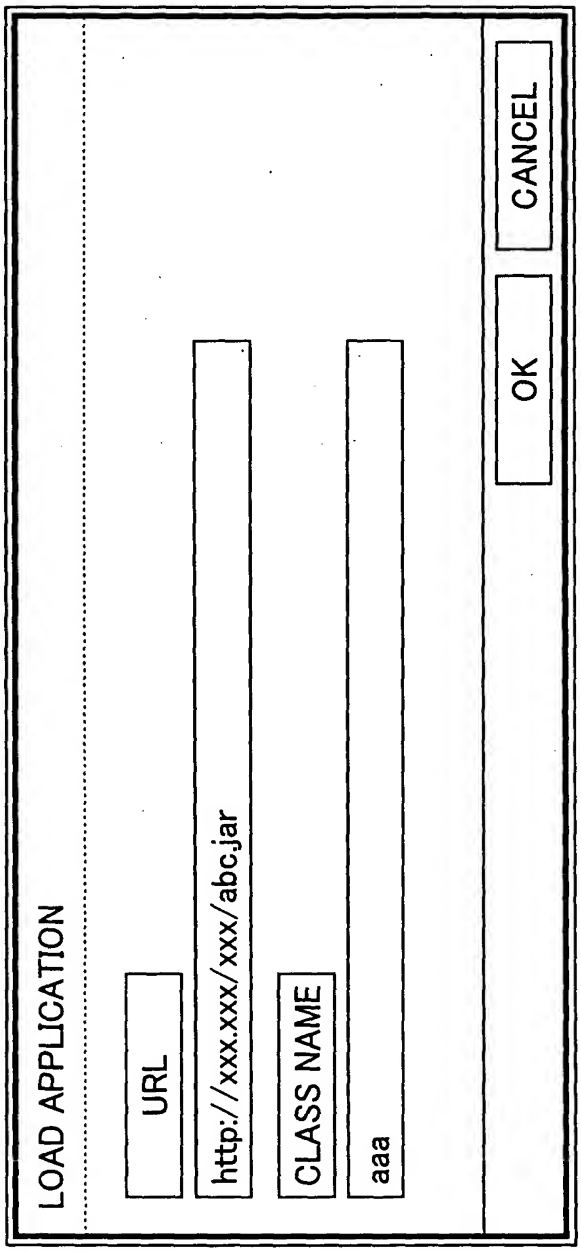

LOAD APPLICATION
URL
http://xxx.xxx/xxx/abc.jar
CLASS NAME
aaa
OK
CANCEL

FIG.42

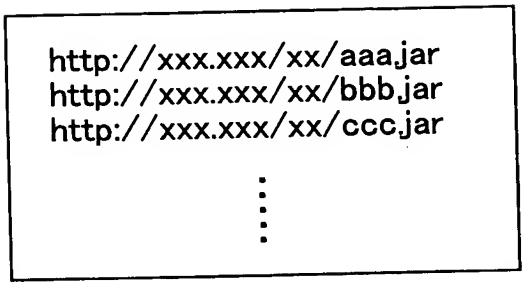

http://xxx.xxx/xx/aaa.jar
http://xxx.xxx/xx/bbb.jar
http://xxx.xxx/xx/ccc.jar

FIG.43

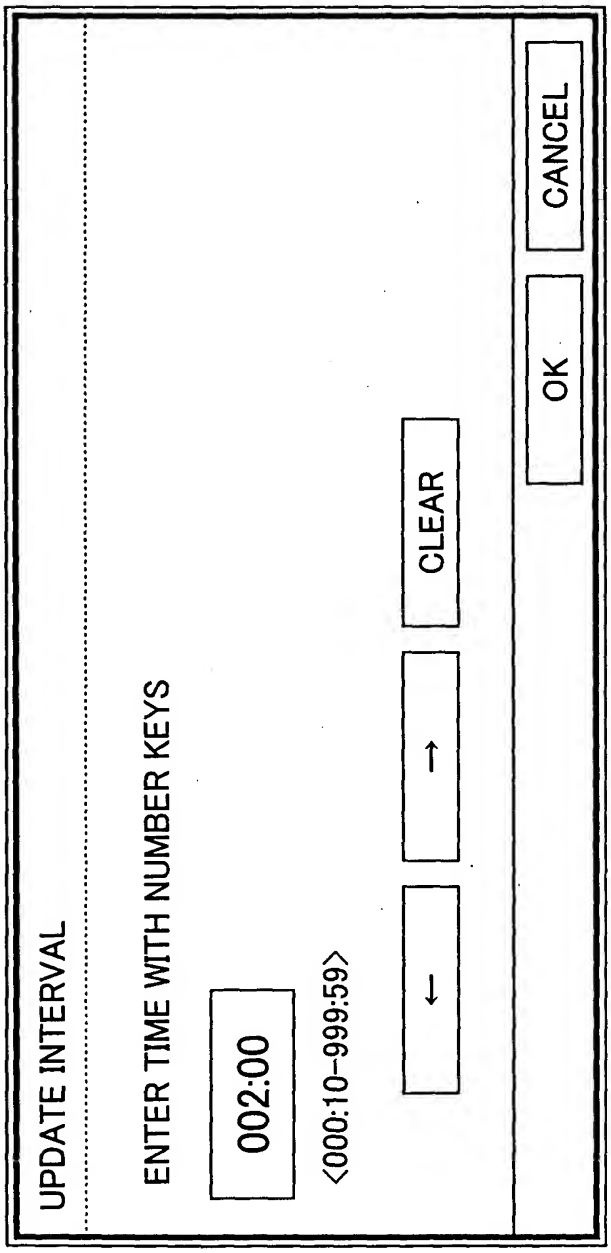

UPDATE INTERVAL
ENTER TIME WITH NUMBER KEYS
002:00
<000:10-999:59>
←
→
CLEAR
OK
CANCEL

FIG.44

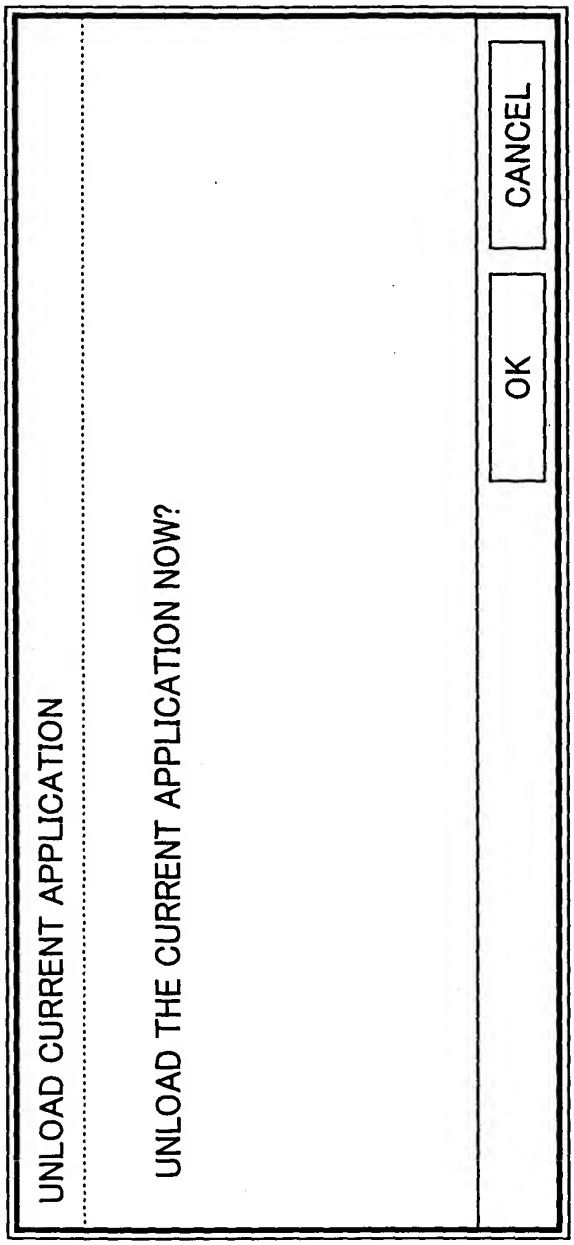

UNLOAD CURRENT APPLICATION
UNLOAD THE CURRENT APPLICATION NOW?
OK
CANCEL

FIG.45

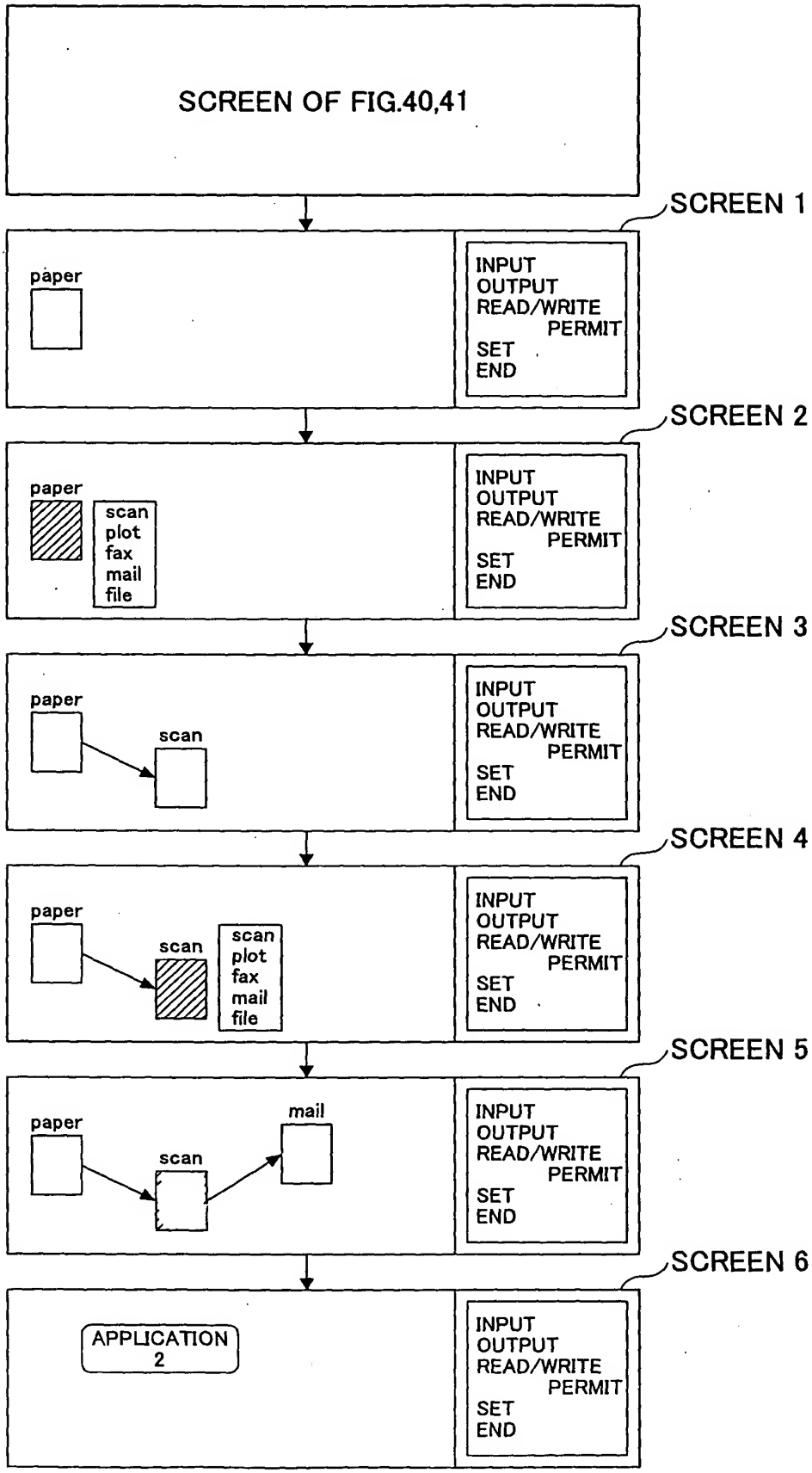

SCREEN OF FIG.40,41
SCREEN 1
paper
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END
SCREEN 2
paper
scan
plot
fax
mail
file
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END
SCREEN 3
paper
scan
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END
SCREEN 4
paper
scan
scan
plot
fax
mail
file
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END
SCREEN 5
paper
scan
mail
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END
SCREEN 6
APPLICATION 2
INPUT
OUTPUT
READ/WRITE
PERMIT
SET
END

FIG.46

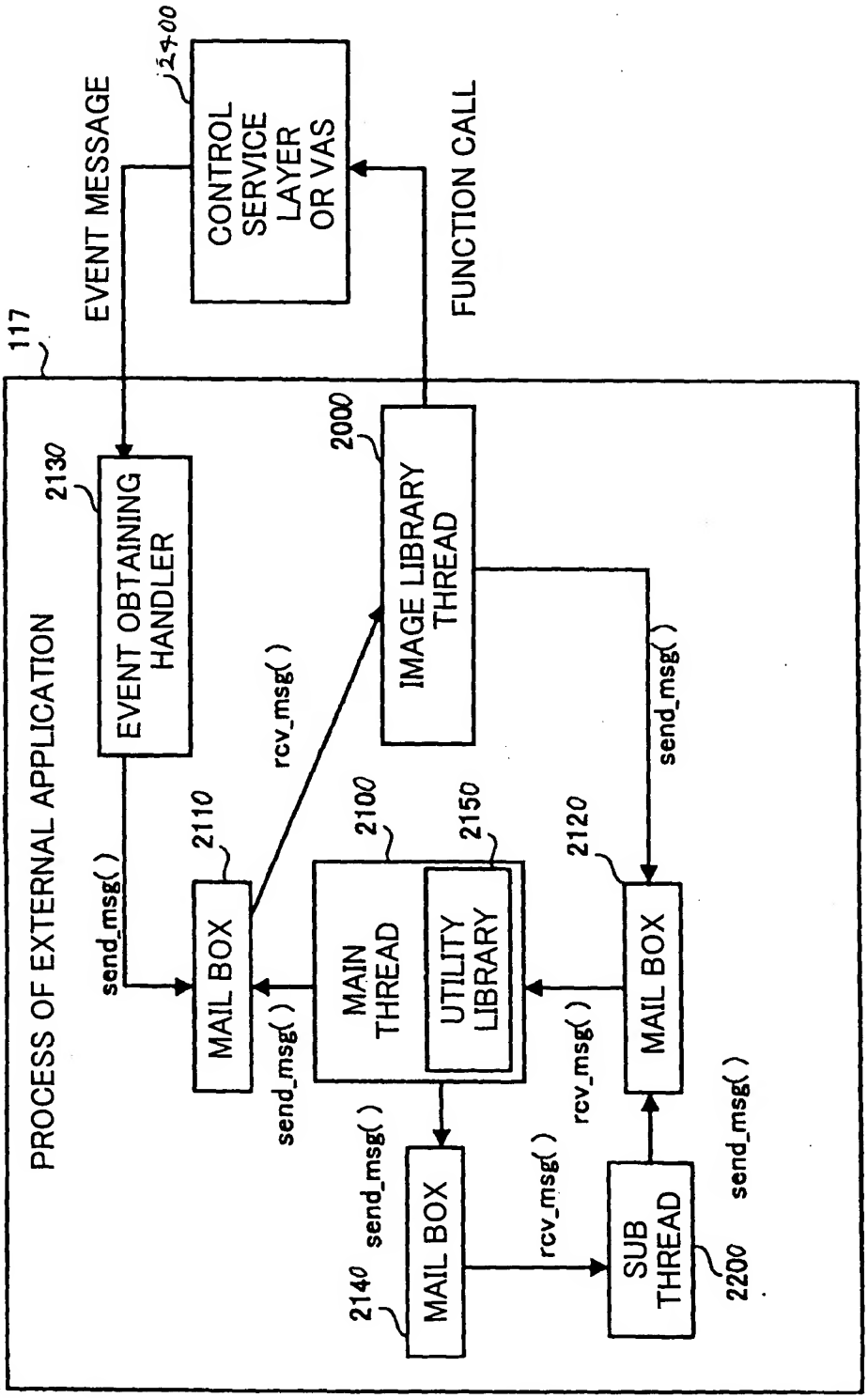

2400
CONTROL SERVICE LAYER OR VAS
EVENT MESSAGE
FUNCTION CALL
117
PROCESS OF EXTERNAL APPLICATION
2130
EVENT OBTAINING HANDLER
2000
IMAGE LIBRARY THREAD
send_msg( )
rcv_msg( )
send_msg( )
2110
2100
2150
2120
MAIL BOX
MAIN THREAD
UTILITY LIBRARY
MAIL BOX
send_msg( )
rcv_msg( )
rcv_msg( )
send_msg( )
send_msg( )
2140
MAIL BOX
rcv_msg( )
SUB THREAD
2200

FIG.47

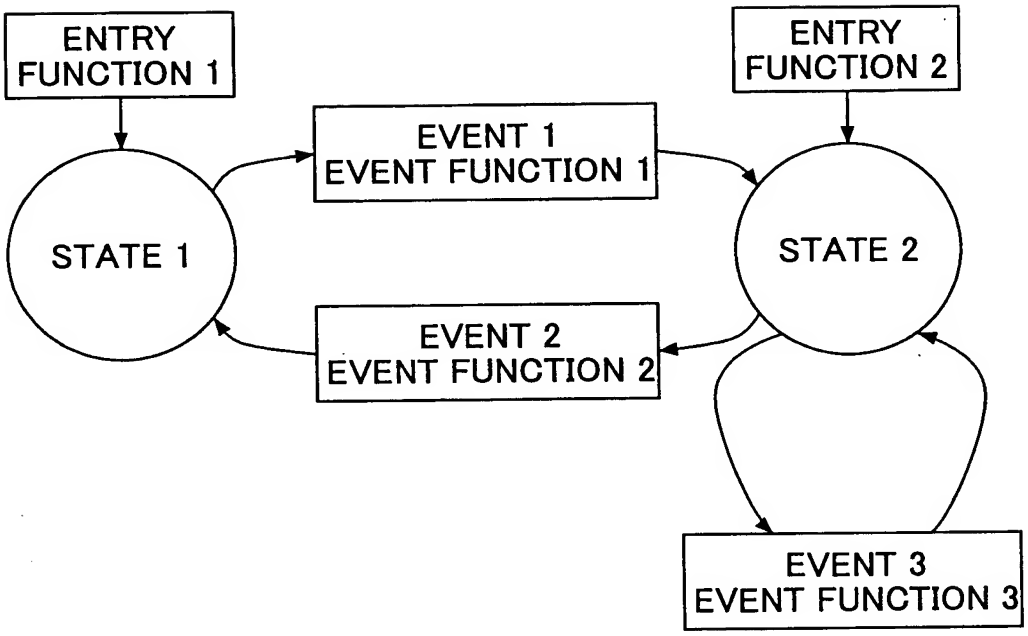

ENTRY FUNCTION 1
ENTRY FUNCTION 2
EVENT 1 EVENT FUNCTION 1
STATE 1
STATE 2
EVENT 2 EVENT FUNCTION 2
EVENT 3 EVENT FUNCTION 3

FIG.48

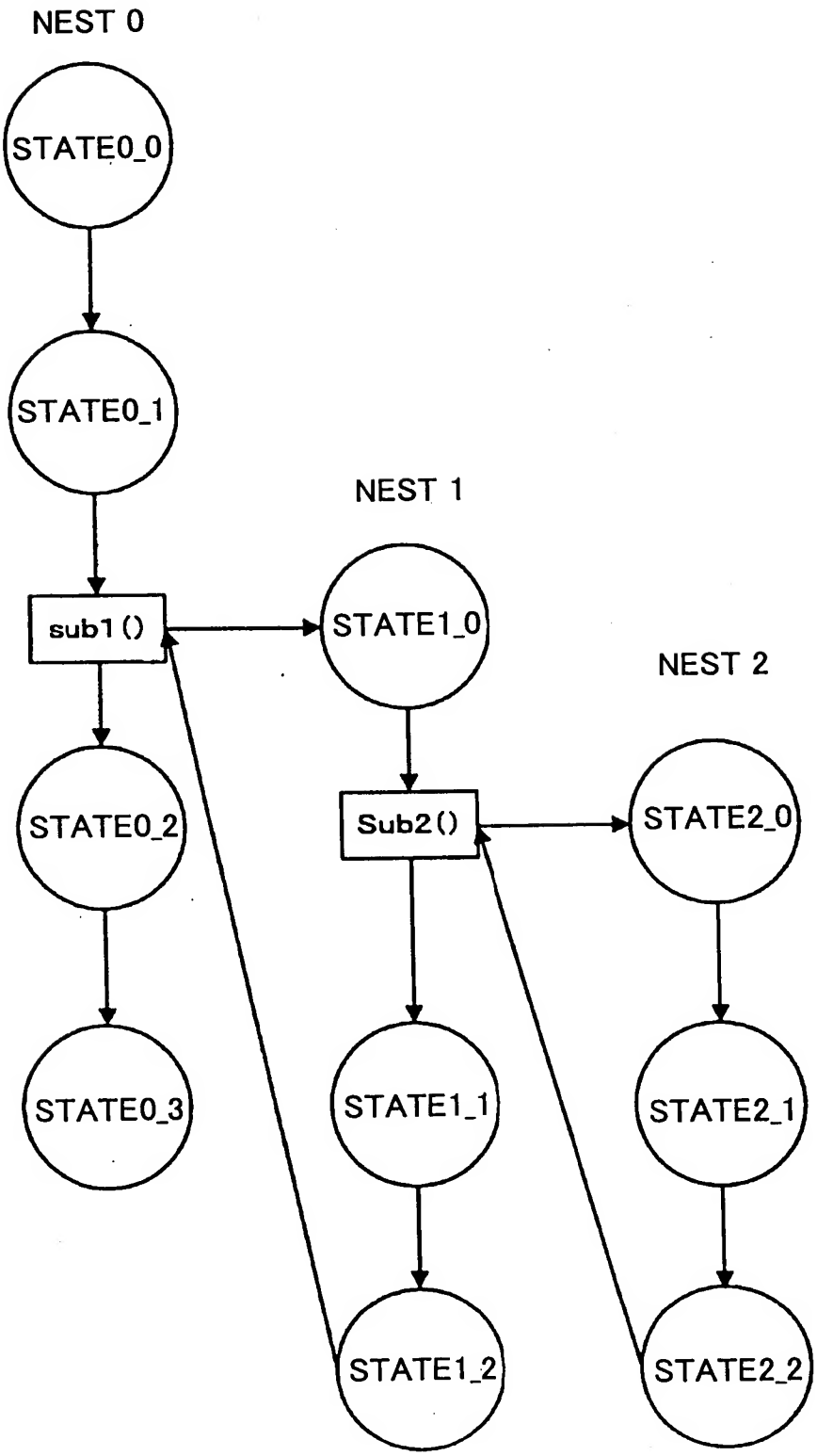

NEST 0
STATE0_0
STATE0_1
NEST 1
sub1 ()
STATE1_0
NEST 2
STATE0_2
Sub2 ()
STATE2_0
STATE0_3
STATE1_1
STATE2_1
STATE1_2
STATE2_2

FIG.49

```
#define  THREAD_NO 1            //THREAD NUMBER :1
typedef struct {
        int     demoData;       //DEFINITION OF SHARED VARIABLE
} demoCommonParm_t;

#undef  COMMON_STRUCT
#define  COMMON_STRUCT demoCommonParm_t // DEFINITION FOR ACCESSING SHARED VARIABLE

enum DEMO_STATE{                // DEFINITION OF STATE
        DEMO_ST_IDLE,
        DEMO_ST_RUN
};

owner TypeFunc_t eventDemoIdle[] = //  EVENT FUNCTION TABLE OF STATE DEMO_ST_IDLE
{
        {       1,      evDemoIdle },
        {       0,      0       }
};

owner TypeFunc_t eventDemoRun[] =  // EVENT FUNCTION TABLE OF STATE DEMO_ST_RUN
{
        {       2,      evDemoRun },
        {       0,      0       }
};

stateInfo_t demoStTbl[] =  // STATE INFORMATION TABLE
{
        {       enterDemoIdle,  eventDemoIdle },
        {       enterDemoRun,  eventDemoRun  }
};
```

OBLON, SPIVAK, ET AL
DOCKET #: 240819US2
INV: Kunihiro AKIYOSHI, et al.
SHEET 42 OF 46

FIG.50

```
/*****************************************************
        execDemoFunc
*****************************************************/
void    execDemoFunc(void)
{
        int ret;
        COMMON_STRUCT *pCom;        //DEFINITION OF SHARED VARIABLE STRUCTURE POINTER
        pCom = (COMMON_STRUCT *)malloc(sizeof(COMMON_STRUCT));
                                    // AREA RESERVE FOR SHARED VARIABLE
        pCom->demoData = 0;    // SET OF SHARED VARIABLE

        ret = StMachine(THREAD_NO, demoStTble, DEMO_ST_IDLE, pCom);  // LAUNCH
        free(pCom);                 // RELEASE OF SHARED VARIABLE
}

/*****************************************************
        enterDemoIdle  ENTRY FUNCTION
*****************************************************/
void    enterDemoIdle(void)
{
        ...
}
/*****************************************************
        evDemoIdle      EVENT FUNCTION
*****************************************************/
void    evDemoIdle(void)
{
        ...
}
/*****************************************************
        enterDemoRun  ENTRY FUNCTION
*****************************************************/
void    enterDemoRun(void)
{
        ...
}
/*****************************************************
        evDemoRun      EVENT FUNCTION
*****************************************************/
void    evDemoRun(void)
{
        ...
}
```

FIG.51

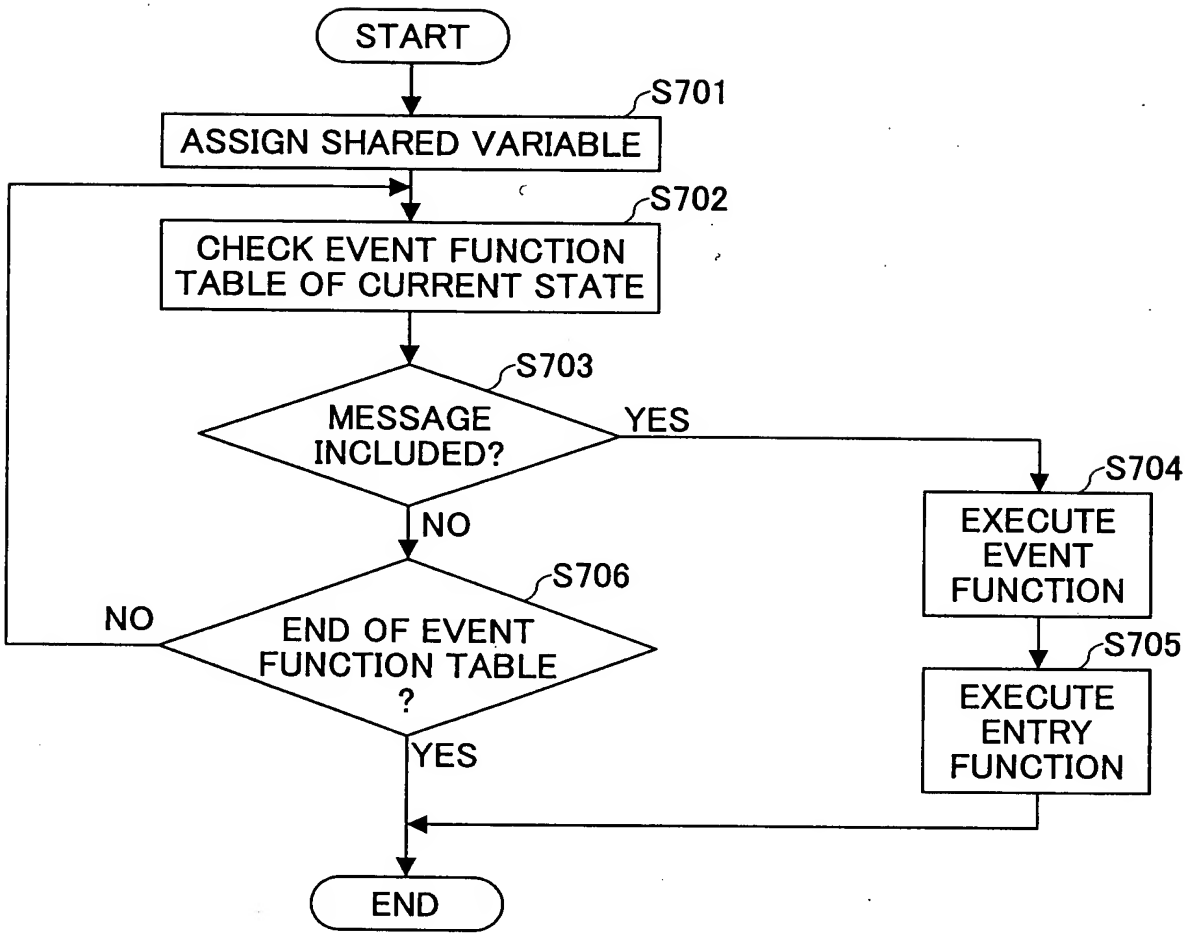

START
S701
ASSIGN SHARED VARIABLE
S702
CHECK EVENT FUNCTION TABLE OF CURRENT STATE
S703
MESSAGE INCLUDED?
YES
NO
S704
EXECUTE EVENT FUNCTION
S705
EXECUTE ENTRY FUNCTION
S706
END OF EVENT FUNCTION TABLE ?
NO
YES
END

FIG.52

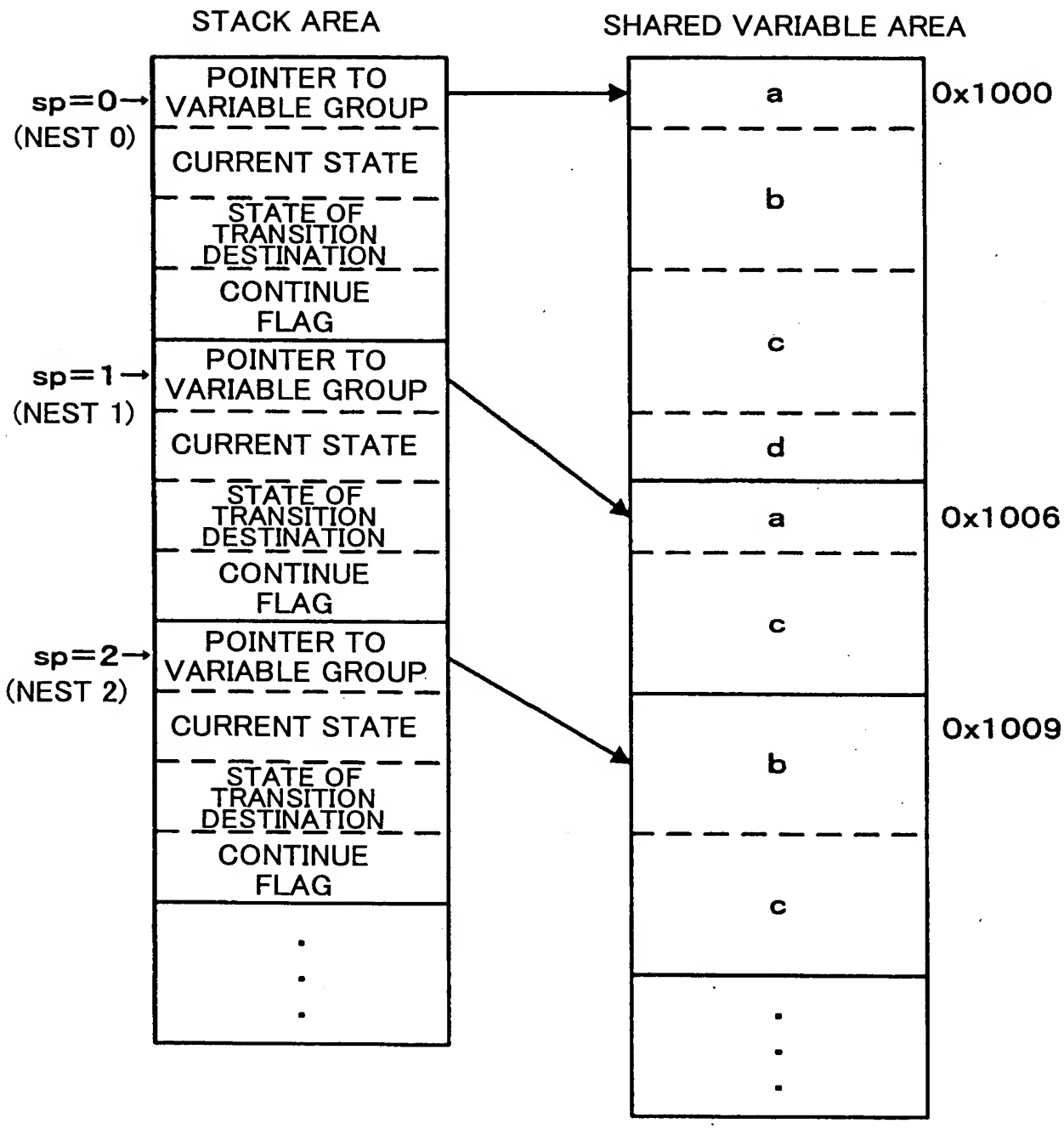
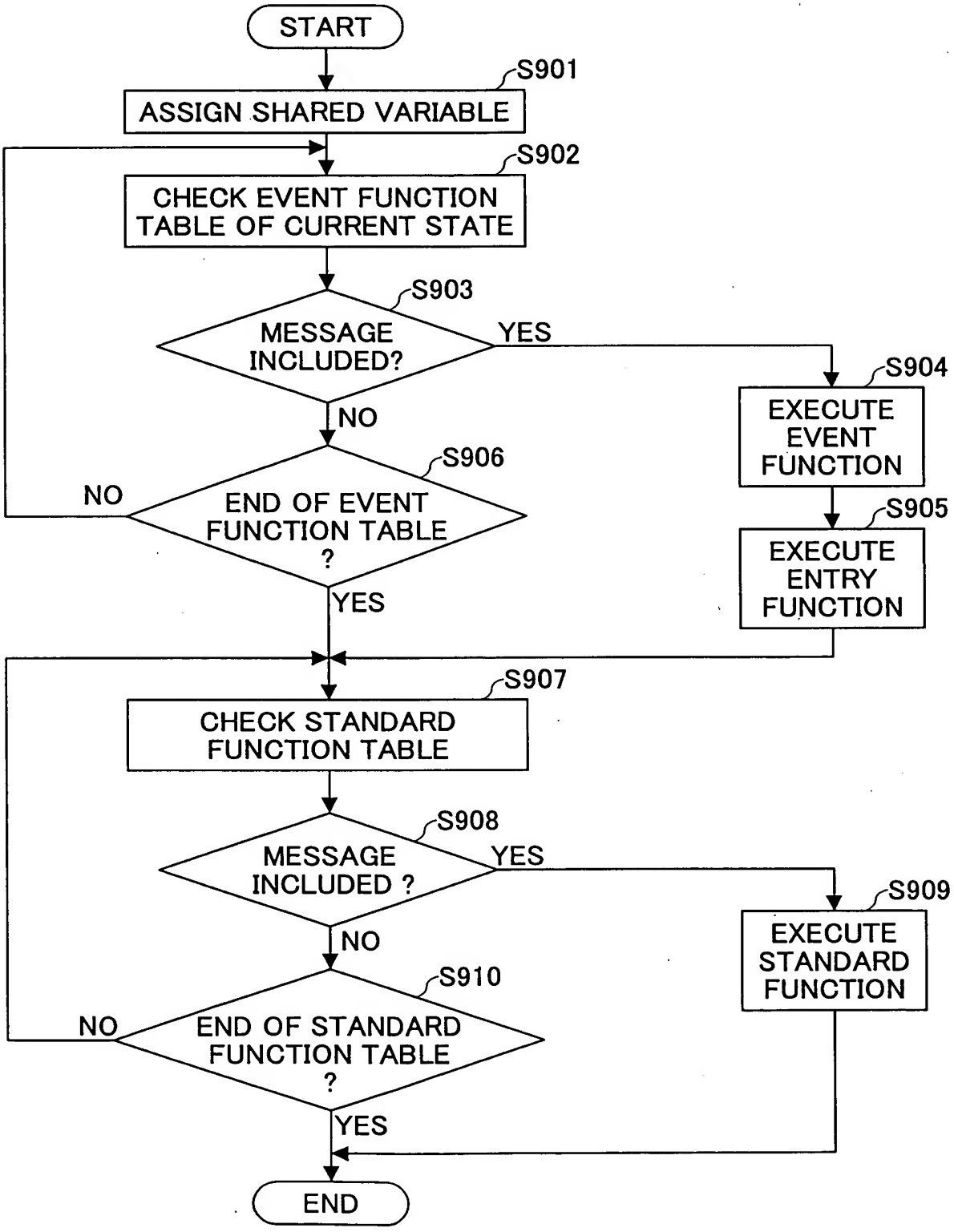

STACK AREA
SHARED VARIABLE AREA
sp=0
(NEST 0)
POINTER TO VARIABLE GROUP
CURRENT STATE
STATE OF TRANSITION DESTINATION
CONTINUE FLAG
sp=1
(NEST 1)
POINTER TO VARIABLE GROUP
CURRENT STATE
STATE OF TRANSITION DESTINATION
CONTINUE FLAG
sp=2
(NEST 2)
POINTER TO VARIABLE GROUP
CURRENT STATE
STATE OF TRANSITION DESTINATION
CONTINUE FLAG
a
b
c
d
0x1000
0x1006
0x1009

FIG.53


START
S901
ASSIGN SHARED VARIABLE
S902
CHECK EVENT FUNCTION TABLE OF CURRENT STATE
S903
MESSAGE INCLUDED?
YES
NO
S904
EXECUTE EVENT FUNCTION
S905
EXECUTE ENTRY FUNCTION
S906
END OF EVENT FUNCTION TABLE ?
NO
YES
S907
CHECK STANDARD FUNCTION TABLE
S908
MESSAGE INCLUDED ?
YES
NO
S909
EXECUTE STANDARD FUNCTION
S910
END OF STANDARD FUNCTION TABLE ?
NO
YES
END